RAMAIAH
Institute of Technology

CURRICULUM

for the Academic year 2019 – 2020

COMPUTER SCIENCE AND ENGINEERING

VII & VIII 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

VII SEMESTER

SI. No	Course Code	Course Name	Category	Credits					Contact Hours
				L	T	P	S*	Total	
1.	CS71	Computer Network Security	PC-C	3	0	0	1	4	42
2.	CS72	High Performance Computing	PC-C	3	0	0	1	4	42
3.	CS73	Data Analytics	PC-C	3	1	0	0	4	42+28
4.	CSExx	Group A	PC-E	*	*	*	*	4	--
5.	CSExx	Group B	PC-E	*	*	*	*	4	--
6.	CSExx	Group C	PC-E	*	*	*	*	4	--
7.	CSL74	High Performance Computing Laboratory	PC-C	0	0	1	0	1	28
8.	CSL75	Data Analytics Laboratory	PC-C	0	0	1	0	1	28
Total				-	-	-	-	26	--

Professional Elective List for VII Semester

	Group A		Group B		Group C	
SI. No	Course Code	Course Name	Course Code	Course Name	Course Code	Course Name
1.	CSE14	Cloud Computing (3:0:1:0)	CSE19	Analysis of Computer Data Networks(4:0:0:0)	CSE24	Project management &Engg Economics(4:0:0:0)
2.	CSE15	Storage Area Networks (4:0:0:0)	CSE20	Distributed Systems (4:0:0:0)	CSE25	Advanced DBMS (3:1:0:0)
3.	CSE16	Network Management(4:0:0:0)	CSE21	Advanced Computer Networks (4:0:0:0)	CSE26	Software Architecture & Design Patterns (4:0:0:0)
4.	CSE17	Pattern Recognition (4:0:0:0)	CSE22	Multimedia Computing(4:0:0:0)	CSE27	Semantic Web (3:1:0:0)
5.	CSE18	Digital Forensics (3:0:1:0)	CSE23	Introduction to Deep Learning (3:0:0:1)	CSE28	Wireless Sensor Networks (4:0:0:0)

VIII SEMESTER

SI. No	Course Code	Course Name	Category	Credits				
				L	T	P	S*	Total
1.	XXOExx	Institutional Elective	OE	4	0	0	0	4
2.	CSIN	Internship/Departmental Elective (Industry collaborated course)	IN or PC-E	*	*	*	*	4
3.	CSP	Project Work	PW	0	0	16	0	16
4.	EAC	Extra-Curricular/Co-Curricular Activities*	EAC	0	0	0	2	2
Total				--	--	--	--	26

Departmental Elective List for VIII Semester

SI. No	Subject Code	Subject
1.	CSE29	Software Testing
2.	CSE30	Industry collaborated course
3.	CSE31	SOA and Web Services

Computer Security

Course Code: CS71 **Credits: 3:0:0:1**
Prerequisites: Nil **Contact Hours: 42L**
Course Coordinator/s: Sowmya B J

Course Contents:

Unit I

Introduction: Security Goals, Cryptographic Attacks, Services and Mechanism, Techniques. Mathematics of Cryptography: Integer Arithmetic, Modular Arithmetic, Matrices.
Self-Study: Linear Congruence.

Unit II

Traditional Symmetric-Key Ciphers: Introduction, Substitution Ciphers, Transpositional Ciphers, Stream and Block Ciphers.
Data Encryption Standard (DES): Introduction, DES Structure, DES Analysis, Security of DES.
Advanced Encryption Standard: Introduction, Transformations, Key Expansion, The AES Ciphers.
Self-Study: Examples, Analysis of AES.

Unit III

Encipherment using Modern Symmetric-Key Ciphers: Use of Modern Block Ciphers, Use of Stream Ciphers, Other Issues.
Asymmetric Key Cryptography: Introduction, RSA Cryptosystem.
Self-Study: Rabin Cryptosystem, Elgamal Cryptosystem.

Unit IV

Message authentication: Authentication Requirements, Authentication Functions, Message Authentication Codes.
Digital signatures: Digital Signatures, Digital Signature Algorithm.
Key management and distribution: Distribution of public keys, X.509 certificates.
Self-Study: Kerberos.

Unit V

System security: Intruders: Intruders, Intrusion detection. **Malicious Software**: Types of Malicious Software, Viruses. Firewalls: The need for Firewalls, Firewall Characteristics.
Self-Study: Types of Firewalls.

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. Behrouz A. Forouzan, Debdeep Mukhopadhyay: Cryptography and Network Security, 2nd Edition, Special Indian Edition, Tata McGraw-Hill, 2011.
2. William Stallings, Cryptography and Network Security, Fifth Edition, Prentice Hall of India, 2005.

Reference Book:

1. Josef Pieprzyk, Thomas Hardjono, Jennifer Serberry Fundamentals of Computer Security, Springer, ISBN 978-3-662-07324-7.

Course Outcomes (COs):

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

1. Interpret security goals and the various threats to security (PO1, PO2, 4,PO6, 12 and *PS02, PS03*)
2. Identify the type of encryption method DES or AES depending on the need and security threat perception (PO1, PO2,3, PO6 and *PS02, PS03*)
3. Compare the need of Symmetric-key and Asymmetric-key Ciphers. (PO1, PO2, PO6 and *PS02, PS03*)
4. Summarize the fundamentals of Key Management and Identity need for Digital Signatures and its utility (PO1, PO2, PO6 and *PS02, PS03*)
5. Appraise the need for firewalls and system security. (PO2, PO6 and *PS02*, *PS0*3)

High Performance Computing

Course Code: CS72 **Credits: 3:0:0:1**
Prerequisites: Computer Organization **Contact Hours: 42L**
Course Coordinator/s: Mallegowda M

Course Contents:

Unit I

Introduction to High–Performance Computers, Memory Hierarchy, CPU Design: Reduced Instruction Set Computers, Multiple–Core Processors, Vector Processors.
Self-Study: Parallel Semantics, Distributed Memory Programming.

Unit II

Programming Shared Address Space Platforms: Thread Basics, Why Threads? The POSIX Thread API, Thread Creation and Termination, Synchronization Primitives in Pthreads, Controlling Thread and Synchronization Attributes, Thread Cancellation, Composite Synchronization Constructs.
Self-Study: Tips for Designing Asynchronous Programs, OpenMP: a Standard for Directive Based Parallel Programming.

Unit III

Programming using the Message-Passing Paradigm: Principles of Message-Passing Programming, The Building Blocks: Send and Receive Operations, MPI: the Message Passing Interface, Topologies and Embedding, Overlapping Communication with Computation, Collective Communication and Computation Operations**.**
Self-Study: Groups and Communicators.

Unit IV

Introduction: GPUs as Parallel Computers, Architecture of a Model GPU, Why More Speed or Parallelism? Parallel Programming Languages and Models, Overarching Goals. History of GPU Computing: Evolution of Graphics Pipelines, GPU Computing. Introduction to CUDA: Data Parallelism, CUDA Program Structure, A Matrix-Matrix Multiplication Example, Device Memories and Data Transfer.
Self-Study: Kernel Functions and Threading.

Unit V

CUDA Threads: CUDA Thread Organization, Using blockIdx and threadIdx, Synchronization and Transparent Scalability, Thread Assignment, Thread Scheduling and Latency Tolerance. CUDA Memories: Importance of Memory Access Efficiency, CUDA Device Memory Types, A Strategy for Reducing Global Memory Traffic.
Self-Study: Memory as a limiting Factor to Parallelism.

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. Rubin H Landau, Oregon State University, http://science.oregonstate.edu/rubin/.
2. AnanthGrama, AnshulGupta, Vipinkumar, George Karypis, Introduction to parallel computing, second edition, 2003, Pearson education publishers.
3. David B Kirk, Wen-mei W. Hwu, "Programming Massively Parallel Processors – A Hands-on Approach", First Edition, Elsevier and nvidia Publishers, 2010.

Reference Books:

1. Thomas Rauber and GudulaRunger Parallel Programming for Multicore and cluster systems,Springer International Edition, 2009.
2. Hennessey and Patterson Computer Architecture: A quantitative Approach, Morgan Kaufman Publishers, 2011.
3. Michael J.Quin "Parallel Programming in C with MPI and Open MP", McGraw Hill.

Course Outcomes (COs):

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

1. Summarise the technologies and architectures used for parallel computing (PO-1,2,3,4,PSO-1,3)
2. Design and develop parallel programs using Open-MP programming interface. (PO-1,2,3,4,PSO-1,3)
3. Apply the principles and architecture of message-passing programming paradigm for solving real world problems(PO-1,2,3,4,PSO-1,3)
4. Outline the characteristics and architecture of Graphical Processing Units (PO-1,2,3,4,PSO-1,3)
5. Design parallel applications using CUDA-C in GPU environments (PO-1,2,3,4,PSO-1,3)

Data Analytics

Course Code: CS73 **Credits: 3:1:0:0**
Prerequisites: Nil **Contact Hours: 42L+28T**
Course Coordinator/s: Srinidhi H

Course Contents:

Unit I

Data Science in a Big world: Benefits and uses of Data Science in Big data, Facets of data, Big data ecosystem and Data Science, Data Science process.

Unit II

Linear Regression: Simple Linear Regression, Multiple Linear Regression, Considerations for linear regression model, Comparison of Linear regression with k-nearest neighbours Classification: Logistic Regression, Linear Discriminant analysis, Comparison of Classification methods.

Unit III

Mining data streams : Introduction To Streams Concepts – Stream Data Model and Architecture - Stream Computing - Sampling Data in a Stream – Filtering Streams – Counting Distinct Elements in a Stream – Estimating Moments – Counting Oneness in a Window – Decaying Window - Real time Analytics Platform(RTAP) Applications – Case Studies - Real Time Sentiment Analysis- Stock Market Predictions.

Unit IV

Analytics Beyond Hadoop: Berkeley Big-data Analytics (BDA) Stack: Motivation, Design and Architecture, Real-time Analytics with Storm, GraphLab: Processing Large Graphs.

Unit V

Data Visualization: Context of Data Visualization, Taxonomy of data visualization methods, Choosing the appropriate chart type. Comparison of Categories: Dot Plots, Bar Charts, Floating bar (or Gantt chart), Pixelated bar chart, Histogram, Slope graph (or bumps chart or table chart), Radial chart, Glyph chart, Sankey diagram, Area size chart, Small multiples (or trellis chart), Word cloud.

Text Books:

1. Cielen, D., Meysman, A., & Ali, M. (2016). Introducing data science: big data, machine learning, and more, using Python tools. Manning Publications Co.
2. James, G., Witten, D., Hastie, T., &Tibshirani, R. (2013). An introduction to statistical learning (Vol. 112). New York: springer.
3. Leskovec, J., Rajaraman, A., & Ullman, J. D. (2014). Mining of massive datasets. Cambridge university press.
4. Agneeswaran, V. S. (2014). Big data analytics beyond hadoop: real-time applications with storm, spark, and more hadoop alternatives. FT Press.
5. "Data Visualization": a successful design process, Kirk, Andy, Packt Publishing Ltd, 2012.

Reference Book:

1. Friedman, J., Hastie, T., &Tibshirani, R. (2001). The elements of statistical learning (Vol. 1, pp. 337-387). New York: Springer series in statistics.

Course Outcomes (COs):

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

1. Interpret the process of data analytics and its significance for projects (PO- 1, 2, 3, 4, 5, PSO- 1, 2, 3)
2. Apply the different methods of data analytics to regression and classification problems (PO – 1, 2, 3, 4, 5, PSO – 2, 3)
3. Summarise the context of streams and different analytical methods on it(PO – 1, 2, 3, 4, 5, PSO – 2, 3)
4. Use different types of analytical tools beyond Hadoop and their applications(PO – 1, 2, 3, 4, 5, PSO – 2, 3)
5. Recognize the need for visualization and its significance in data analytics(PO – 1, 2, 3, 4, 5, PSO – 2, 3)

High Performance Computing Laboratory

Course Code: CSL74 **Credits: 0:0:1:0**
Prerequisites: FOC, OOPs **Contact Hours: 28P**
Course Coordinator/s: Dr. Anita Kanavalli

Course Contents:

1. Programs on #pragma using C.
2. Programs using Sections, omp for and omp single.
3. Programs using thread private directives.
4. Programs on scheduling.
5. Programs using last private reduction, copyin and shared.
6. Programs for Point to Point MPI calls.
7. Programs for Message passing MPI calls.
8. Programs on CUDA.

Text Books:

1. Parallel Programming in OpenMP ,Rohit Chandra , Leo Dagum , DrorMaydan , David Kohr, Jeff McDonald , Ramesh Menon.
2. Multi-core programming,Increase performance through software multi-yhreading by Shameem Akhter and Jason Roberts.

Course Outcomes (COs):

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

1. Propose HPC solutions to real time problems (PO- 1, 2, 3, 5, PSO-2)
2. Develop programs using OpenMP, MPI and CUDA (PO- 1, 2, 3, 5, PSO-2)
3. Examine the performance of OpenMP, MPI and CUDA programs (PO- 1, 2, 3, 4, 5, PSO-2)

Data Analytics Laboratory

Course Code: CSL75 **Credits: 0:0:1:0**

Contact Hours: 28P

Prerequisites: C programming/C++/Python, Data mining/Machine learning

Course Coordinator/s: Srinidhi H

Course Contents:

Laboratory on R programming and it's applications in data analytics are conducted based on the following topics:

1. Introduction to R: Why use R? Working with R.Reading and writing data: Vectors, Matrices, Arrays, Data frames, Factors & Lists in R.
2. Basic data management: Creating new variables, Missing values, sorting data, merging datasets, sub setting datasets.
3. Basic graphs: Pie charts, scatterplot, histogram, box plots, dot plots with in-built datasets in R.
4. Advanced data management: User-written functions, mathematical functions, statistical functions, applying functions on matrices and data frames.
5. Basic Statistics: Descriptive statistics and its methods, Frequency and contingency tables, Correlations, t-tests.
6. Intermediate methods: Linear regression, analysis of variance in R.

Text Books:

1. EMC Education Services. (2015). Data Science and Big Data Analytics: Discovering, Analysing, Visualizing and Presenting Data. John Wiley & Sons.
2. Robert I. Kabacoff, (2011). R in Action. Manning.

Reference Books:

1. Spector, P. (2008). Data manipulation with R. Springer Science & Business Media.
2. Matloff, N. (2011). The art of R programming: A tour of statistical software design. No Starch Press.

Course Outcomes (COs):

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

1. Understand the different types of data and its structures. (PO2, PO3, PO5, PSO2)
2. Analyse the different types of data pre-processing techniques involved in Management of data. (PO2, PO3, PO5, PSO2)
3. Demonstrate the usage of graphical analysis of data, different classes of analytical techniques both basic and advance statistics using R and need for data analytical applications (PO1, PO2, PO3, PO5,PO12, PSO3)

Cloud Computing

Course Code: CSE14 **Credits: 3:0:1:0**
Prerequisites: FOC, OOPs **Contact Hours: 42L+28P**
Course Coordinator/s: Dr. S Rajarajeswari

Course Contents:

Unit I

Introduction: Network centric computing and network centric content, Peer-to-peer systems, Cloud Computing: an old idea whose time has come, Cloud Computing delivery models & Services, Ethical issues, Cloud vulnerabilities, Major Challenges, **Cloud Infrastructure:** Amazon, Google, Azure & online services, open source private clouds. Storage diversity and vendor lock-in, intercloud, Energy use & ecological impact of data centers, Service and Compliance level agreement, Responsibility sharing, User experience, Software licensing.

Unit II

Cloud Computing: Applications & Paradigms, Challenges, existing and new application opportunities, Architectural styles of cloud applications; different cloud architectures: single , multi ,hybrid cloud site, redundant, non-redundant , 3 tier, multi-tier architectures; Workflows coordination of multiple activities, Coordination based on a state machine model -the Zoo Keeper, The Map Reduce programming model, Apache Hadoop, A case study: the GrepTheWeb application, **Applications**: Healthcare, Energy systems, transportation, manufacturing, Education, Government, mobile communication, application development.

Unit III

Cloud Resource Virtualization: Layering and virtualization, Virtual machine monitors, Virtual machines, VM Performance and security isolation, Full virtualization and paravirtualization, Application, Desktop, hardware, software, network, storage, memory Virtualization, Hardware support for virtualization Case study: Xen -a VMM based on paravirtualization, Optimization of network virtualization in Xen 2.0, vBlades -paravirtualization targeting a x86-64 Itanium processor, A performance comparison of virtual machines, Virtual machine security, The darker side of virtualization, Software fault isolation.

Unit IV

Cloud Resource Management and Scheduling: Policies and mechanisms for resource management, Applications of control theory to task scheduling on a cloud, Stability of a two-level resource allocation architecture, Feedback control based on dynamic thresholds, Coordination of specialized autonomic performance managers, A utility-based model for cloud-based web services, Resource bundling, combinatorial auctions for cloud Scheduling algorithms for

computing clouds, fair queuing, Start time fair queuing, borrowed virtual time, Cloud scheduling subject to deadlines, Scheduling mapreduce applications subject to deadlines, Resource management and application scaling.

Unit V

Storage systems: Evolution, Storage models, file systems, databases, DFS, General parallel File system, GFS, Hadoop, Locks & Chubby, TPS, NOSQL, Bigdata, Mega stosre.

Cloud Security: Risks, privacy and privacy impacts assessments, Trust, OS, VM security, security of virtualization, risk posed by shared images, mgmt OS, Xoar, Trusted VMM.

Text Books:

1. Cloud Computing Theory and Practice – DAN C. Marinescu – Morgan Kaufmann Elsevier.
2. Cloud Computing A hands - on approach – ArshdeepBahga& Vijay madisetti Universities press.

Reference Books:

1. Cloud Computing: Theory and Practice, Dan Marinescu, 1st edition, MK Publishers, 2013.
2. Distributed and Cloud Computing, From Parallel Processing to the Internet of Things, Kai Hwang, Jack Dongarra, Geoffrey Fox. MK Publishers.
3. Cloud Computing: A Practical Approach, Anthony T. Velte, Toby J. Velte, Robert Elsenpeter, McGraw Fill, 2010.
4. Internet – White papers from IBM, other enterprise resources.
5. https://docs.rightscale.com/cm/designers_guide/cm-cloud Computing-system-architecture-diagrams.html.

Course Outcomes (COs):

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

1. Justify the need of new technology of Cloud Computing and its ecological impact (PO- 4, 5, 6, 7, PSO-3)
2. Design own cloud Architectural model, workflow and programming model for real time applications (PO- 1, 2, 3, 4, PSO-3)
3. Develop Hardware architectures for virtualizing the resources as Cloud service provider (PO- 1, 2, 3, 4, PSO-3)
4. Develop new policies and mechanism for Cloud Resource Management and Scheduling.(PO- 2, 4, PSO-3)
5. Assess the risk and security issues involved with the cloud computing Environment (PO- 3, 4, PSO-3)

List of Lab Exercises:

Use Eucalyptus or Open Nebula or equivalent to set up the cloud and demonstrate:

1. SaaS: Google Drive
   With your Gmail account, create a spreadsheet to share with the people at the same table, invite them.
   See how you can simultaneously edit the document you just created
   PaaS: google Maps
   http://maps.google.com/maps/api/staticmap?center=Eiffel+Tower&zoom=12&size=512x512&sensor=false.
2. Design Virtual Machine using VM player and test Client server application using KVM,Vmware..
3. Design Virtual Machine using VM player and test Client server application using Virtual Box.
4. Iaas: Study and implementation of Infrastructure as a Service (Iaas) by installing openstack using Quanta plus/Aptana/Kompozer technology.
5. Paas – Deploy Applications to google App Engine - simple web applications and with database.
6. Write a program to illustrate single –sign- on , an open ID by installing and using JOSSO.
7. Deploy Applications to cloud foundry using VMC, Micro cloud foundry, Eclipse.
8. Write a program for webfeed and RSS to illustrate the concept of form and control validation using PHP, HTML
9. Write program to secure servers in cloud by installing and using security feature of ownCloud.
10. To Set up a Hadoop Cluster – Single Node, Multi Node and Execute Map Reduce Programs in Hadoop Cluster.
11. Case study on AWS - students will learn about Amazon EC2. Amazon Elastic Compute Cloud is a central part of Amazon.com's cloud computing platform, Amazon Web Services. EC2 allows users to rent virtual computers on which to run their own computer applications.
12. Case study on Microsoft Azure- students will learn about Microsoft Azure is a cloud computing platform and infrastructure, created by Microsoft, for building, deploying and managing applications and services through a global network of Microsoft-managed datacenters. How it work, different services provided by it.
13. Saas: installing ownCloud : An enterprise file sharing solution for online collaboration and storage.
14. Implementation of identity management using Openstack features.

Storage Area Networks

Course Code: CSE15 **Credits: 4:0:0:0**
Contact Hours: 56L

Prerequisites: Computer Networks, Computer Organization, Operating Systems

Course Coordinator/s: Dr. Divakar Harekal

Course Contents:

Unit I

Introduction: Information Storage, Evolution of Storage Architecture, Data Centre Infrastructure, Virtualization and Cloud Computing. Data Centre Environment: Application, DBMS, Host, Connectivity, Storage, Disk Drive Components, Disk Drive Performance, Host Access to Data, Direct-Attached Storage, Storage Design Based on Application, Disk Native Command Queuing, Introduction to Flash Drives.

Unit II

Data Protection: RAID Implementation Methods, Array Components, Techniques, Levels, Impact on Disk Performance, Comparison, Hot Spares. Intelligent Storage System: Components, Storage Provisioning, Types.

Unit III

Fiber Channel Storage Area Networks: FC Overview, Evolution, Components, FC Connectivity, Ports, FC Architecture, Fabric Services, Login Types, Zoning, FC Topologies, Virtualization in SAN.IP SAN and FCoE: iSCSI, FCIP, FCoE.

Unit IV

Network-Attached Storage: Benefits, Components, NAS I/O Operation, Implementations, File Sharing Protocols, I/O Operations, Factors Affecting NAS Performance, File-Level Virtualization Object Based and Unified Storage: Object Based Storage Devices, Content Addressed Storage, CAS Use Cases, Unified Storage. Backup Archive and Replication.

Unit V

Business Continuity: Information Availability, Terminology, Planning Lifecycle, Failure Analysis, Impact Analysis, Challenges, Adoption Considerations. Securing the Storage Infrastructure: Framework, Risk Triad, Domains Managing the Storage Infrastructure: Monitoring, Management Activities, Management Challenges, Information Lifecycle Management, Storage Tiering.

Text Book:

1. EMC Education Services, edited by Somasundaram G., Alok Shrivastava "Information Storage and Management"; 2nd edition , Wiley India, 2012, ISBN 9788126537501.

Reference Books:

1. Ulf Troppens, Rainer Erkens and Wolfgang Muller: Storage Networks Explained, 1st Edition, Wiley India, 2012.
2. Robert Spalding: Storage Networks, The Complete Reference, 1st Edition, Tata McGraw Hill, 2011.

Course Outcomes (COs):

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

1. Identify the need for storage centric network and its benefits of its adoption (PO- 1, 2, 3, 4, PSO-1,2)
2. Design a storage solution for an application depending on the IOPS and RAID requirements (PO- 1, 2, 3, 4, PSO-1,2)
3. Understand the Fiber channel stack and working of the different layers (PO- 1, 2, 3, 4, PSO-1,2)
4. Summarize about NAS, object oriented storage and backup and recovery (PO- 1, 2, 3, 4, PSO-1,2)
5. Develop a business continuity plan and ILM of an enterprise (PO- 1, 2, 3, 4, PSO-1,2)

Network Management

Course Code: CSE16 **Credits: 4:0:0:0**
Prerequisites: Computer Networks **Contact Hours: 56L**
Course Coordinator/s: Dr. Monica R Mundada

Course Contents:

Unit I

Data communications and Network Management Overview: Analogy of Telephone Network Management, Communications protocols and Standards, Case Histories of Networking and Management, Challenges of Information *T*echnology Managers, Network Management: Goals, Organization, and Functions, Network and System Management, Network Management System Platform, Current Status and future of Network Management.

Unit II

SNMPV1 Network Management: Organization and Information Models, Managed network: Case Histories and Examples, The History of SNMP Management, The SNMP Model, The Organization Model, System Overview, The Information Model. SNMPv1 Network Management: Communication and Functional Models. The SNMP Communication Model, Functional Model.

Unit III

SNMP Management: SNMPv2: Major Changes in SNMPv2, SNMPv2 System Architecture, SNMPv2 Structure of Management Information, SNMPv2 Management Information Base, SNMPv2 Protocol, Compatibility With SNMPv1. SNMP Management: SNMPv3,Introduction,SNMPv3 Key Features,SNMPv3 Documentation Architecture, SNMPv3 Applications, SNMPv3 Management Information Base, Security, SNMPv3 User-based Security Model, Authentication Protocols, Encryption Protocol**.**

Unit IV

SNMP Management: RMON: What is Remote Monitoring? , RMON SMI and MIB, RMON1, RMON2, ATM Remote Monitoring, a Case Study of Internet Traffic Using RMON, ATM Networks.

Unit V

Network Management Tools and Systems: Network Management Tools, Network Statistics Measurement Systems, History of Enterprise Management, Network Management systems, Commercial Network management Systems, System Management, Enterprise Management Solutions.

Text Book:

1. Network Management: Principles and Practice, Mai Subramanian, George institute of Technology, 2000, Pearson.

Course Outcomes (COs):

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

1. Identify five major functional areas of network management. (PO-1,3,4, PSO- 1, 2)
2. Explore SNMP Management Information, Standard MIBs and SNMP protocol. (PO- 1,3,4, PSO- 1, 2)
3. Defend the need for security in networking monitoring and control. (PO-1,3,4, PSO- 1, 2)
4. Devise Remote network monitoring System for statistics collection, alarms and filters. (PO-1,3,4, PSO- 1, 2)
5. Summarize complete network management plan for a moderate to large network enterprise through case studies. (PO-1,3,4, PSO- 1, 2)

Pattern Recognition

Course Code: CSE17 **Credits: 4:0:0:0**
Prerequisites: Artificial Intelligence **Contact Hours: 56L**
Course Coordinator/s: Dr. Annapurna P Patil

Course Contents:

Unit I

Introduction: Machine perception, an example, Pattern Recognition System, The Design Cycle, Learning and Adaptation. Bayesian Decision Theory: Introduction, Bayesian Decision Theory, Continuous Features, Minimum error rate, classification, classifiers, discriminant functions, and decision surfaces, the normal density, Discriminant functions for the normal density.

Unit II

Maximum-likelihood and Bayesian Parameter Estimation: Introduction, Maximum-likelihood estimation, Bayesian Estimation, Bayesian parameter estimation: Gaussian Case, general theory, Hidden Markov Models. Non-parametric Techniques: Introduction, Density Estimation, Parzen windows, KN – Nearest- Neighbor Estimation, The Nearest- Neighbor Rule, Metrics and Nearest-Neighbor Classification.

Unit III

Linear Discriminant Functions: Introduction, Linear Discriminant Functions and Decision Surfaces, Generalized Linear Discriminant Functions, The Two-Category Linearly Separable case, Minimizing the Perception Criterion Functions, Relaxation Procedures, Non-separable Behavior, Minimum Squared-Error procedures, The Ho-Kashyap procedures. Stochastic Methods: Introduction, Stochastic Search, Boltzmann Learning, Boltzmann Networks and Graphical Models, Evolutionary Methods.

Unit IV

Non-Metric Methods: Introduction, Decision Trees, CART, Other Tree Methods, Recognition with Strings, Grammatical Methods.

Unit V

Unsupervised Learning and Clustering: Introduction, Mixture Densities and Identifiability, Maximum-Likelihood Estimates, Application to Normal Mixtures, Unsupervised Bayesian Learning, Data Description and Clustering, Criterion Functions for Clustering.

Text Book:

1. Richard O. Duda, Peter E. Hart, and David G. Stork: Pattern Classification, 2nd Edition, Wiley-Interscience, 2012.

Reference Book:

1. Earl Gose, Richard Johnson baugh, Steve Jost: Pattern Recognition and Image Analysis, HAR/DSK Edition, Pearson Education, 2007.

Course Outcomes (COs):

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

1. Analyze using Top-down approach the pattern recognition System (PO-1, 2, PSO-2)
2. Interpret Bayesian decision theorem(PO-1, 2, PSO-2)
3. Analyze the Bayesian estimation, Density estimation. (PO-1, 2, PSO-2)
4. Explain linear discriminant functions(PO-1, 2, PSO-2)
5. Use different stochastic methods for linear discriminant functions, assess the criteria for unsupervised learning and clustering (PO-1, 2, PSO-2)

Digital Forensics

Course Code: CSE18 **Credits: 3:0:1:0**
Prerequisites: Nil **Contact Hours: 42L+28P**
Course Coordinator/s: Dr. S Ramani

Course Contents:

Unit I

Understanding Cyber Crime: Indian IT Act 2008 and amendments, Computer Forensic and Investigations as a Profession, Understanding Computer Forensics. **Understanding Computer Investigations:** Preparing a Computer Investigation, Taking a Systematic Approach, Procedures for Corporate High-Tech Investigations, Understanding Data Recovery Workstations and Software.

Unit II

Working with Windows and DOS Systems: Understanding File Systems, Exploring Microsoft File Structures, Examining NTFS Disks, Understanding Whole Disk Encryption, Understanding the Windows Registry, Understanding Microsoft Startup Tasks, Understanding MS-DOS Startup Tasks, and Understanding Virtual Machines.

Unit III

Data Acquisition: Understanding Storage Formats for Digital Evidence, Determining the best Acquisition Method, Contingency Planning for Image Acquisitions, Using Acquisition Tools, Validating Data Acquisitions, Using Remote Network Acquisition Tools. **Computer Forensics Analysis and Validation:** Determining What Data to Collect and Analyze, Validating Forensic Data, Addressing Data-Hiding Techniques, Performing Remote Acquisitions.

Unit IV

Current Computer Forensics Tools: Evaluating Computer Forensic Tool Needs, Computer Forensics Software Tools, Computer Forensics Hardware Tools, Validating and Testing Forensics Software. **Recovering Graphics Files:** Recognizing a Graphics File, Understanding Data Compression, Locating and Recovering Graphics Files, Identifying Unknown File Formats, Understanding Copyright Issues with Graphics.

Unit V

Network Forensics: Network Forensic Overview, Performing Live Acquisitions, Developing Standard Procedures for Network Forensics, Using Network Tools. **E-mail Investigations:** Exploring the Role of E-mail in

Investigations, Exploring the Roles of the Client and Server in E-mail, Investigating E-mail Crimes and Violations, Understanding E-mail Servers, Using Specialized E-mail Forensics Tools. Laboratory Lab exercises using forensic software and Case study data.

Text Book:

1. Nelson, Phillips, Frank, Enfinger and Steuart: Computer Forensics and Investigations, Cengage Learning, 2008. (Chapters: 1, 2, 4, 6, 7, 8, 9, 10, 11, 12)

Reference Books:

1. Marjie T. Britz: Computer Forensics and Cyber Crime - An Introduction, 2nd Edition, Pearson Education, 2012.
2. Harish Chander: Cyber Laws and IT Protection, PHI, 2012.
3. http://www.cyberforensics.in/default.aspx

List of Lab Exercises:

The following exercises have to be performed using various open source software tools/utilities mentioned.

Software Tools:

1. CyberCheck 4.0 - Academic Version
2. CyberCheckSuite
3. MobileCheck
4. Network Session Analyser
5. Win-LiFT
6. TrueImager
7. TrueTraveller
8. PhotoExaminerVer 1.1
9. CDRAnalyzer

Course Outcomes (COs):

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

1. Recall the Indian IT Act 2008 and its amendments. (PO1,3,4, PSO3)
2. Classify various types of computer crime. (PO1,3,4, PSO3)
3. Apply computer forensic techniques to identify the digital fingerprints associated with criminal activities. (PO1,3,4, PSO3)
4. Analyze hidden information from pictures and other files. (PO1,3,4, PSO3)
5. Apply Network Forensic tools for network forensic and live data forensic analysis. (PO1,3,4, PSO3)

Analysis of Computer Data Networks

Course Code: CSE19 **Credits: 4:0:0:0**
Prerequisites: Computer Networks, Data Communication **Contact Hours: 56L**
Course Coordinator/s: Mamatha Jadhav V

Course Contents:

Unit I

Review of Digital Transmission, Error Detection, Effectivness of Error Detection codes, Two-dimensional Parity checks, Polynomial codes. Standardized Polynomial Codes, Error Detecting Capability of a Polynomial code, Linear Codes, Performance of Linear Codes, Error Correction.

Unit II

Introduction to Data Link Layer, Framing, Character – oriented Framing, Bit-oriented Framing, Length fields, Framing with errors, Maximum Frame size-Variable Frame Length, Fixed Frame Length, Little's theorem, Probabilistic Form of Little's theorem, Applications of Little's theorem, Occupancy distribution upon arrival, Occupancy Distribution upon arrival, Occupancy distribution up to departure, Brief review of Queuing models.

Unit III

Delay Models in Data Networks – Introduction, Multiplexing of Traffic on a Communication Link, Queuing Models. Statistical Multiplexing, Poisson arrival process and packet loss probability, M/M/m/m: The m-server loss system, M/G/1 system, Priority Queuing, non-preemptive priority, Preemptive resume priority, Network of queues, Jackson's theorem.

Unit IV

Multi-access communication-Review of Aloha Networks, Idealized Slotted Multi-access Model, Stabilized Slotted Aloha, Splitting algorithms, Tree Algorithms, FCFS Splitting Algorithms, Improvements in the FCFS splitting Algorithm, CSMA Slotted Aloha, FCFS Splitting Algorithms for CSMA.

Unit V

Topological Design of Networks: Flow Models, An overview of Topological Design Problems, Subnet Design Problem, Capacity Assignment Problem, Heuristic Methods for Capacity Assignment, Network Reliability Issues, Spanning Tree Topology Design.

Text Books:

1. Alberto Leon-Garcia and IndraWidjaja, Communication Networks, Fundamental Concepts and Key Architectures, Second Reprint, Tata McGraw-Hill, 2004.
2. Dimitri Bertsekas and Robert Gallager, Data Networks, Second Edition, Prentice Hall of India, 2000.

Reference Books:

1. Anurag Kumar, Manjunath and Joy Kuri: Communication Networking.
2. An Analytical Approach, Indian Reprint, Morgan Kaufman Publishers, 2006.

Course Outcomes (COs):

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

1. Illustrate mathematical and engineering aspects of the performance in Data Networks. (PO-1,2, PSO-1)
2. Create a mathematical model of the given problem in computer Networks.(PO-1,3,4,5, PSO-1)
3. Identify the role of Little's theorem in the analysis typically in Occupancy distribution on arrival and Occupancy Distribution on Departure.(PO-1,2,3,4,5, PSO-1)
4. Review the study on Queuing models and apply to the cases of Priority Queuing, non-preemptive .(PO-1,3,5, PSO-1)
   priority and preemptive resume priority
5. Describe special theorems like Jackson's theorem and its applications, splitting algorithms and application to ALOHA Networks, design problems (PO-1,2,3,4,5,PSO-1)

Distributed Systems

Course Code: CSE20 **Credits: 4:0:0:0**
Prerequisites: OS **Contact Hours: 56L**
Course Coordinator/s: Sini Anna Alex

Course Contents:

Unit I

Introduction : Definition, Relation to computer system components, Motivation, Relation to parallel multiprocessor/multicomputer systems, Message-passing systems versus shared memory systems, Primitives for distributed communication, Synchronous versus asynchronous executions, Design issues and challenges.

A model of distributed computations: A distributed program, A model of distributed executions, Models of communication networks, Global state of a distributed system, Cuts of a distributed computation, Past and future cones of an event, Models of process communications

Logical time: Introduction, A framework for a system of logical clocks, Scalar time, Vector time, Efficient implementations of vector clocks, Jard–Jourdan's adaptive technique, Matrix time, Virtual time, Physical clock synchronization: NTP.

Unit II

Global state and snapshot recording algorithms: Introduction, System model and definitions, Snapshot algorithms for FIFO channels, Variations of the Chandy–Lamport algorithm, Snapshot algorithms for non-FIFO channels, Snapshots in a causal delivery system, Monitoring global state, Necessary and sufficient conditions for consistent global snapshots, Finding consistent global snapshots in a distributed computation.

Terminology and basic algorithms: Topology abstraction and overlays, Classifications and basic concepts, Complexity measures and metrics, Program structure, Elementary graph algorithms, Synchronizers, Maximal independent set (MIS), Connected dominating set, Compact routing tables, Leader election, Challenges in designing distributed graph algorithms, Object replication problems.

Unit III

Message ordering and group communication: Message ordering paradigms, Asynchronous execution with synchronous communication, Synchronous program order on an asynchronous system, Group communication, Causal order (CO), Total order, A nomenclature for multicast, Propagation trees for multicast, Classification of application-level multicast algorithms, Semantics of fault-tolerant group communication, Distributed multicast algorithms at the network layer, .

Termination detection: Introduction, System model of a distributed computation, Termination detection using distributed snapshots, Termination detection by weight throwing, A spanning-tree-based termination detection algorithm, Message-optimal termination detection, Termination detection in a very general distributed computing model, Termination detection in the atomic computation model, Termination detection in a faulty distributed system.

Unit IV

Distributed mutual exclusion algorithms: Introduction, Preliminaries, Lamport's algorithm, Ricart–Agrawala algorithm, Singhal's dynamic information-structure algorithm, Lodha and Kshemkalyani's fair mutual exclusion algorithm, Quorum-based mutual exclusion algorithms, Maekawa's algorithm, Agarwal–El Abbadi quorum-based algorithm, Token-based algorithms, Suzuki–Kasami's broadcast algorithm, Raymond's tree-based algorithm,

Deadlock detection in distributed systems: Introduction, System model, Preliminaries, Models of deadlocks, Knapp's classification of distributed deadlock detection algorithms, Mitchell and Merritt's algorithm for the single resource model, Chandy–Misra–Haas algorithm for the AND model, Chandy–Misra–Haas algorithm for the OR model, Kshemkalyani–Singhal algorithm for the P-out-of-Q model.

Unit V

Global predicate detection: Stable and unstable predicates, Modalities on predicates, Centralized algorithm for relational predicates, Conjunctive predicates, Distributed algorithms for conjunctive predicates, Further classification of predicates.

Consensus and agreement algorithms: Problem definition, Overview of results, Agreement in a failure-free system (synchronous or asynchronous), Agreement in (message-passing) synchronous systems with failures, Agreement in asynchronous message-passing systems with failures, Wait-free shared memory consensus in asynchronous systems.

Peer-to-peer computing and overlay graphs: Introduction, Data indexing and overlays, Unstructured overlays, Chord distributed hash table, Content addressable networks (CAN), Tapestry, Some other challenges in P2P system design, Tradeoffs between table storage and route lengths, Graph structures of complex networks, Internet graphs, Generalized random graph networks, Small-world networks, Scale-free networks, Evolving networks.

Text Book:

1. Ajay D. Kshemkalyani, and MukeshSinghal "Distributed Computing: Principles, Algorithms, and Systems", Cambridge University Press, 2008 (Reprint 2013).

Reference Books:

1. John F. Buford, Heather Yu, and Eng K. Lua, "P2P Networking and Applications", Morgan Kaufmann, 2009 Elsevier Inc.
2. Kai Hwang, Geoffrey C. Fox, and Jack J. Dongarra, "Distributed and Cloud Computing: From Parallel processing to the Internet of Things", Morgan Kaufmann, 2012 Elsevier Inc.

Course Outcomes (COs):

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

1. Identify the design issues and Challenges in building distributed systems. (PO3, PSO-1)
2. Explore different ways of managing time (clock) and recording global state of distributed computation. (PO-1,2, 3, PSO-2))
3. Analyze basic distributed graph algorithms, synchronizers, and practical graph problems, P2P overlay problems(PO-2,3,4, PSO-2)
4. Discuss ways to achieve various message ordering schemes and approaches for detecting termination of a distributed computation. (PO-1, 2, 3, PSO-2)
5. Identify different assertion based, and tree based distributed algorithms to implement Distributed Mutual Exclusion. (PO-1,2,3, 4, PSO-2)

Advanced Computer Networks

Course Code: CSE21 **Credits: 4:0:0:0**
Prerequisites: Computer Networks **Contact Hours: 56L**
Course Coordinator/s: Dr. Shilpa S Choudhari

Course Contents:

Unit I

Foundation: Building a Network, Requirements, Perspectives, Scalable Connectivity, Cost-Effective Resource sharing, Support for Common Services, Manageability, Protocol layering, Performance, Bandwidth and Latency, Delay X Bandwidth Product, Perspectives on Connecting, Classes of Links, Reliable Transmission, Stop-and-Wait, Sliding Window, Concurrent Logical Channels.

Unit II

Internetworking I: Switching and Bridging, Datagram's, Virtual Circuit Switching, Source Routing, Bridges and LAN Switches, Basic Internetworking (IP), What is an Internetwork?, Service Model, Global Addresses, Datagram Forwarding in IP, sub netting and classless addressing, Address Translation (ARP), Host Configuration (DHCP), Error Reporting (ICMP), Virtual Networks and Tunnels.

Unit III

Internetworking- II: Network as a Graph, Distance Vector (RIP), Link State (OSPF), Metrics, The Global Internet, Routing Areas, Routing among Autonomous systems (BGP), IP Version 6 (IPv6), Mobility and Mobile IP.

Unit IV

End-to-End Protocols: Simple Demultiplexer (UDP), Reliable Byte Stream(TCP), Endto-End Issues, Segment Format, Connecting Establishment and Termination, Sliding Window Revisited, Triggering Transmission, Adaptive Retransmission, Record Boundaries, TCP Extensions, Queuing Disciplines, FIFO, Fair Queuing, TCP Congestion Control, Additive Increase/ Multiplicative Decrease, Slow Start, Fast Retransmit and Fast Recovery.

Unit V

Congestion Control and Resource Allocation: Congestion-Avoidance Mechanisms, DEC bit, Random Early Detection (RED), Source-Based Congestion Avoidance. The Domain Name System (DNS), Electronic Mail (SMTP,POP,IMAP,MIME), World Wide Web (HTTP), Network Management (SNMP).

Reference Books:

1. Larry Peterson and Bruce S Davis "Computer Networks :A System Approach" 5th Edition , Elsevier -2014.
2. Douglas E Comer, "Internetworking with TCP/IP, Principles, Protocols and Architecture" 6th Edition, PHI – 2014.
3. Uyless Black, "Computer Networks, Protocols , Standards and Interfaces" 2 nd Edition - PHI.
4. Behrouz A Forouzan, "TCP /IP Protocol Suite" 4 th Edition – Tata McGraw-Hill.

Course Outcomes (COs):

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

1. Explain basic computer network technology. (PO1,2,3,4, PSO-1,2)
2. Explain Data Communications System and its components. (PO1, 2,3,4, PSO-1,2)
3. Identify the different types of network topologies and protocols. (PO1,2,3,4,PSO-1,2)
4. Identify the different types of network devices and their functions within a network. (PO1,2,3,4,PSO1)
5. Illustrate the skills of subnet and routing mechanisms. (PO1,2,3,4,PSO-1))

Multimedia Computing

Course Code: CSE22 **Credits: 4:0:0:0**
Prerequisites: Nil **Contact Hours: 56L**
Course Coordinator/s: Aparna R

Course Contents:

Unit I

Introduction, Media and Data Streams, Audio Technology: Multimedia Elements, Multimedia Applications, Multimedia Systems Architecture, Evolving Technologies for Multimedia Systems, Defining Objects for Multimedia Systems, Multimedia Data Interface Standards, The need for Data Compression, Multimedia Databases. Media: Perception Media, Representation Media, Presentation Media, Storage Media, Characterizing Continuous Media Data Streams. Sound: Frequency, Amplitude, Sound Perception and Psychoacoustics, Audio Representation on Computers, Three Dimensional Sound Projection, Music and MIDI Standards, Speech Signals, Speech Output, Speech Input, Speech Transmission. Graphics and Images, Video Technology, Computer-Based Animation: Capturing Graphics and Images Computer Assisted Graphics and Image Processing, Reconstructing Images, Graphics and Image Output Options. Basics, Television Systems, Digitalization of Video Signals, Digital Television, Basic Concepts, Specification of Animations, Methods of Controlling Animation, Display of Animation, Transmission of Animation, Virtual Reality Modeling Language.

Unit II

Data Compression: Storage Space, Coding Requirements, Source, Entropy, and Hybrid Coding, Basic Compression Techniques, JPEG: Image Preparation, Lossy Sequential DCT-based Mode, Expanded Lossy DCT-based Mode, Lossless Mode, Hierarchical Mode. H.261 (Px64) and H.263: Image Preparation, Coding Algorithms, Data Stream, H.263+ and H.263L, MPEG: Video Encoding, Audio Coding, Data Stream, MPEG-2, MPEG-4, MPEG-7, Fractal Compression.

Unit III

Optical Storage Media: History of Optical Storage, Basic Technology, Video Discs and Other WORMs, Compact Disc Digital Audio, Compact Disc Read Only Memory, CD-ROM Extended Architecture, Further CD-ROM-Based Developments, Compact Disc Recordable, Compact Disc Magneto-Optical, Compact Disc Read/Write, Digital Versatile Disc. Content Analysis: Simple Vs. Complex Features, Analysis of Individual Images, Analysis of Image Sequences, Audio Analysis, Applications.

Unit IV

Data and File Format Standards: Rich-Text Format, TIFF File Format, Resource Interchange File Format (RIFF), MIDI File Format, JPEG DIB File Format for Still and Motion Images, AVI Indeo File Format, MPEG Standards, TWAIN.

Unit V

Multimedia Application Design: Multimedia Application Classes, Types of Multimedia Systems, Virtual Reality Design, Components of Multimedia Systems, Organizing Multimedia Databases, Application Workflow Design Issues, Distributed Application Design Issues.

Text Books:

1. Ralf Steinmetz, Klara Narstedt: Multimedia Fundamentals: Vol 1-Media Coding and Content Processing, First Edition, PHI, 2010.
2. Prabhat K. Andleigh, Kiran Thakrar: Multimedia Systems Design,1st Edition, PHI, 2011.

Reference Books:

1. K.R. Rao, Zoran S. Bojkovic and Dragorad A. Milovanovic: Multimedia Communication Systems: Techniques, Standards, and Networks, 1st Edition, PHI , 2010.
2. Nalin K Sharad: Multimedia information Networking, PHI, 2002.

Course Outcomes (COs):

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

1. Identify the basic concepts of media, data streams and audio technology.(PO-1, PSO1)
2. Implement different data compression techniques including video, audio and fractal compression.(PO-1,2, PSO2)
3. Demonstrate different optical storage media including content Analysis.(PO-1,2,3,4, PSO2)
4. Identify the different data and file format standards like TIFF, RIFFF, MIDI and MPEG.(PO-1,3, PSO1)
5. Analyze multimedia application design methods like Virtual Reality design and workflow design.(PO-1,2,3,4,5,PSO2)

Introduction to Deep Learning

Course Code: CSE23 **Credits: 3:0:0:1**
Prerequisites: AI and Machine Learning **Contact Hours: 42L**
Course Coordinator/s: Srinidhi H

Course Contents:

Unit I

Introduction: Human brain, neuron models, neural nets as directed graphs, feedback, neural architectures, knowledge representation, connection to artificial intelligence.
Self-Study: Pytorch and Tensorflow.

Unit II

Learning Process: Error-correction learning, memory based learning, Hebbian learning, competitive learning, Boltzmann learning, credit assignment, learning with and without a teacher, learning tasks, memory, statistical learning theory.
Self-Study: Backpropagation using MNSIT.

Unit III

Modern practical deep neural networks: Deep feedforward networks, regularization for deep learning, optimization for training deep models, convolutional Networks.
Self-Study: Classification using Tensorflow and Pytorch.

Unit IV

Sequence Modelling: Recurrent and recursive nets, practical Methodology, applications.
Self-Study: Reinforcement learning.

Unit V

Deep Learning Research: Linear factor models, auto encoders, variational auto encoders, restricted Boltzmann machine, generative adversarial networks.
Self-Study: Transfer learning.

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. Simon Haykin, Neural networks: A comprehensive foundation, Second Edition, Prentice Hall, New Delhi, 1999, ISBN-81-203-2373-4.
2. Ian Goodfellow, YoshuaBengio and Aaron Courville, Deep Learning, MIT Press, 2016.

Course Outcomes (COs):

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

1. Explain the concepts and applications of neural networks and deep learning. (PO-1,2,3, PSO-2,3)
2. Explain how various types of learning work and how they can be used. (PO-1,2,3, PSO-2,3)
3. Apply deep feedforward networks and convolutional to solve practical problems. (PO-2,3,4,5,PSO-2,3)
4. Demonstrate how recurrent and recursive nets function and how practical problems can be mapped to them. (PO-1,2,3,4,5,PSO-2,3)
5. Design end-to-end deep learning architectures involving various types of feedforward networks, auto encoders, RBM, and generative adversarial networks for practical applications. (PO-2,3,4,5, PSO-2,3)

Project Management and Engineering Economics

Course Code: CSE24 **Credits: 4:0:0:0**
Prerequisites: Nil **Contact Hours: 56L**
Course Coordinator/s: Dr. T N R Kumar

Course Contents:

Unit I

Introduction to Engineering Economics: Engineering Decision Makers, Engineering and Economics, Economics: A Capsule View, Problem Solving and Decision Making.
Time Value of Money: Interest and the Time Value of Money, Reasons for Interest, Simple Interest, Compound Interest, Time Value Equivalence, Compound Interest Factors, Cash Flow Diagrams, Calculation of Time Value Equivalences.
Present Worth Comparisons: Conditions for Present Worth Comparisons, Basic Present Worth Comparison Patterns, Comparison of Assets that have unequal lives, Comparison of Assets assumed to have infinite lives.

Unit II

Present Worth Comparisons: Comparison of deferred investments, Future worth comparisons, Valuation, Payback Comparison Method. Equivalent Annual Worth Comparisons: Utilization of Equivalent Annual Worth Comparisons, Consideration of Asset Life, Use of a sinking fund, Equivalent uniform payments when interest rates vary, Annuity contract for a guaranteed income.

Unit III

Rate of Return Calculations: Rate of Return, Minimum Acceptable rate of return, internal rate of return, Consistency of IRR with other economic comparison methods, IRR Misconceptions, Final comments on theory and practice behind interest rates.
Introduction to Project Management: What is project and project management? Role of project manager, a system view of project management, project phases and project cycle, Context of IT projects.
Strategic Planning and Project Selection: Preliminary scope statements, project management plans, project execution, monitoring and control of project work.

Unit IV

Project scope management: what is project scope management? Scope planning and scope management plan, scope definition and project scope statement, creating work breakdown structure, scope verification.

Project time management: importance of project schedules, activity definition, sequencing, resource estimation, duration estimation, schedule development, schedule control
Project cost management: Cost estimation, budgeting, control.

Unit V

Project quality management: Importance of quality management, what is quality management, planning, assurance, control, tools and techniques for quality control.
Project communication management: Importance, communication planning, information distribution.
Project risk management: what is risk management, risk management planning, common source of risk in IT, risk identification, risk monitoring and control**.**

Text Books:

1. James L Riggs, David D Bedworth, Sabah U Randhawa: Engineering Economics, Fourth Edition, TMH, 1996.
2. Kathy Schwalbe: Project Management in IT, India edition, Cengage Learning, 2007.

Reference Books:

1. R. Panneerselvam: Engineering Economics, PHI Learning Pvt. Ltd., 2001.
2. Bob Hughes, Mike Cotterell: Software Project Management, Tata McGraw Hill, 2006.
3. Pankaj Jalote: Software Project Management in Practice, Pearson, 2006.

Course Outcomes (COs):

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

1. Describe the basic concepts of engineering economics and Time Value Equivalence of money. (PO-1,2,3,4,11,12, PSO-2)
2. Calculate present worth, future worth and equivalent annual worth of investments and compare investment alternatives. (PO-1,5,7,9,12, PSO-2)
3. Identify the various rates of returns (PO-3,4,5,7,9,10,12, PSO-2)
4. Estimate the time, scope and cost of a software project. (PO-1,3,5, 6, 8, 9, 10, 11, 12, PSO-2)
5. Identify various quality issues, communication issues and risks in a software project. (PO-3,4,6,7,9,10,11,12, PSO-2)

Advanced Database Systems

Course Code: CSE25 **Credits: 3:1:0:0**
Prerequisites: DBMS **Contact Hours: 42L+28T**
Course Coordinator/s: Dr. Seema S

Course Contents:

Unit I

Database models and overview of Relational data model: Database models: Flat data model, Entity relationship model, Relation model, Record base model, Network model, Hierarchical model, Object oriented data model, Object relation model, Semi structured model, Associative model, Context data model, graph data model, Overview of relational data model, querying relational model with relational algebra, transaction processing and concurrency control.

Unit II

Indexing and Hashing: Basic Concepts, Ordered Indices, B+-Tree Index Files, B+-Tree Extensions, Multiple-Key access, Static Hashing, Dynamic Hashing, Comparison of Ordered Indexing and Hashing, Bitmap Indices, Index Definition in SQL.

Unit III

Query Processing and Optimization: Overview, Measures of Query Cost, Selection Operation, Sorting, Join Operation, Other Operations, Evaluation of Expressions **Query Optimization:** Transformation of Relational Expressions, Estimating Statistics of Expression Results, Choice of Evaluation Plans.

Unit IV

Parallel Databases: Introduction, I/O Parallelism, Inter-query Parallelism, Intra-query Parallelism, Intra-operation Parallelism, Interoperation Parallelism, Query Optimization, Design of Parallel Systems, Parallelism on Multi-core Processors.

Unit V

Distributed Databases: Homogeneous and Heterogeneous Databases, Distributed Data Storage, Distributed Transactions, Commit Protocols, Concurrency Control in Distributed Databases, Availability, Distributed Query Processing, Heterogeneous Distributed Databases, Cloud-Based Databases, Directory Systems.

Text Books:

1. Database System Concepts" by Abraham Silberschatz , Henry F. Korth, sixth Edition ,McGraw Hill Education.
2. Fundamentals of Database Systems" by Ramez Elmasri, Shamkant B. Navathe ,Fifth Edition, Pearson Publications.

Reference Book:

1. Database Management Systems" by Raghu Ramakrishnan, Johanners Gehrke, Second Edition. McGraw-Hill Education

Course Outcomes (COs):

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

1. Summarise different database models and overview of relation database model (PO-1, PSO-1, 2)
2. Understand the methods of storing, managing and interrogating complex data and the background processes involved in query processing (PO- 1, 3, PSO-1, 2)
3. Analyze background methods to optimizing the query (PO- 1, 3, PSO- 1, 2)
4. Understand the concept of distributed databases and parallel databases (PO-1, 3, PSO-1)
5. Analyze the processing of queries in distributed and parallel databases (PO- 1, 3, PSO-1)

Software Architecture and Design Patterns

Course Code: CSE26 **Credits: 4:0:0:0**
Prerequisites: Nil **Contact Hours: 56L**
Course Coordinator/s: Mallegowda M

Course Contents:

Unit I

Introduction: The Architecture Business Cycle: Where do architectures come from? Software processes and the architecture business cycle, What makes a "good" architecture? What software architecture is and what it is not, Other points of view, Architectural patterns, reference models and reference architectures, Importance of software architecture, Architectural structures and views.

Unit II

Architectural Styles and Case Studies: Architectural styles, Pipes and filters, Data abstraction and object-oriented organization, Event-based, implicit invocation, Layered systems, Repositories, Interpreters, Process control, Other familiar architectures, Heterogeneous architectures. Case Studies: Keyword in Context, Instrumentation software, Mobile robotics, Cruise control, Three vignettes in mixed style.

Unit III

Quality: Functionality and architecture, Architecture and quality attributes, System quality attributes, Quality attribute scenarios in practice, Other system quality attributes, Business qualities, Architecture qualities. Achieving Quality: Introducing tactics, Availability tactics, Modifiability tactics, Performance tactics, Security tactics, Testability tactics, Usability tactics, Relationship of tactics to architectural patterns, Architectural patterns and styles.

Unit IV

Architectural Patterns: Introduction, From mud to structure: Layers, Pipes and Filters, Blackboard. Distributed Systems: Broker, Interactive Systems: MVC, Presentation-Abstraction- Control. Adaptable Systems: Microkernel, Reflection.

Unit V

Some Design Patterns: Structural decomposition: Whole – Part, Organization of work: Master – Slave, Access Control: Proxy. Designing and Documenting Software Architecture: Architecture in the life cycle, Designing the architecture, Forming the team structure, Creating a skeletal system, Uses of architectural documentation, Views, Choosing the relevant views, Documenting a view, Documentation across views.

Text Books:

1. Len Bass, Paul Clements, Rick Kazman: Software Architecture in Practice, 2nd Edition, Pearson Education, 2011.
2. Frank Buschmann, Regine Meunier, Hans Rohnert, Peter Sommerlad, Michael Stal: Pattern-Oriented Software Architecture, A System of Patterns, Volume 1, John Wiley and Sons, 2011.
3. Mary Shaw and David Garlan: Software Architecture- Perspectives on an Emerging Discipline, Prentice-Hall of India, 2007.

Reference Books:

1. E. Gamma, R. Helm, R. Johnson, J. Vlissides: Design Patterns- Elements of Reusable Object-Oriented Software, 1st Edition, Pearson Education, 2012.
2. Web site for Patterns: http://www.hillside.net/patterns/.

Course Outcomes (COs):

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

1. Describe the foundational concepts of software architecture and the important principles and techniques of software architecture. (PO-2, 3, 5, PSO-1, 2)
2. Identify the structure, advantages and disadvantages of various architectural choices using case studies. (PO-2, 3, 5, PSO-1, 2)
3. Summarise the need of Software architecture and quality requirements for a software system. (PO-2, 3, 5, 9, PSO-1, 2)
4. Assess different architecture styles and solutions. (PO-2, 3, 5, 9, PSO-1, 2)
5. Apply different architectural views and various design patterns for different software systems. (PO-2, 5, 9, PSO-1, 2)

Semantic Web

Course Code: CSE27 **Credits: 3:1:0:0**
Prerequisites: Web Technologies **Contact Hours: 42L+28T**
Course Coordinator/s: Dr. J Geetha

Course Contents:

Unit I

The basics of semantic web: Traditional web to semantic web – WWW and its usage- meta data and its creation, addition in the web page; meta data tools - search engines for semantic web –search engine for web page mark up problem and query building problem.

Unit II

Resource description frame work (rdf): RDF and its basic elements-Why we need RDF-RDF triples-RDF tools Fundamental rules of RDF- relationship between DC, and RDF and XML and RDF core elements of RDF- ontology and taxonomy-inferencing based on RDF.

Unit III

Web ontology language (owl): The basics idea of Web ontology language– OWL to define classes- OWL to define properties-set operators-Three faces of OWL-Ontology Matching and Distributed Information- Validating OWL ontology.

Unit IV

Semantic web services: Web services – web services standards – web services to semantic web services- UDDI and its usage- Concept of OWL-S and its building blocks - mapping OWL-S to UDDI- WSDL-S overview and its usage.

Unit V

Real world examples and applications of semantic web: Swoogle- architecture, usage and examples of using Swoogle; FOAF – Explanation, vocabulary – creating FOAF documents – overview of semantic markup – semantic web search engines.

Text Book:

1. LiyangYu , “Introduction to the Semantic Web and Semantic web services” Chapman & Hall/CRC, Taylor & Francis group, 2007.

Reference Books:

1. Johan Hjelm, “Creating the Semantic Web with RDF “ , Wiley, 2001
2. Grigoris Antoniou and Frank van Harmelen, “A Semantic Web Primer”, MIT Press, 2004. 234 CS-Engg&Tech-SRM-2013
3. Karin K. Breitman K., Marco Antonio Casanova, Walt Truszkowski, “Semantic web : concepts, Technologies and applications” Walt Truszkowski - 2007.

Course Outcomes (COs):

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

1. Understand the concepts of semantic web technology and appreciate RDF and its taxonomy and ontology (PO- 1, 2,5, PSO-1)
2. Describe OWL and its usage in semantic web (PO- 1, 2,5, PSO-1)
3. Understand various technologies related to semantic web services(PO- 1, 2, 5,PSO-1)
4. Create reusable formal models, and processes to create/update/query such models that help to describe formal semantics used in a multimedia application (PO- 1, 2, 5,PSO-1)
5. Create ontology's in RDFS, including classes and subclasses, properties and sub properties, domains and ranges, instances, and facets, and determine resulting inference and querying capabilities (PO- 1, 2, 5,PSO-1)

Wireless Sensor Networks

Course Code: CSE28 **Credits: 4:0:0:0**
Prerequisites: Computer Networks **Contact Hours: 56L**
Course Coordinator/s: Sanjeetha R

Course Contents:

Unit I

Introduction: Definition of Wireless Sensor Networks (WSNs), difference between the adhoc and sensor n/ws, challenges for WSN and applications of WSN. Single node architecture: Hardware components, energy consumption in sensor nodes, brief study of operating systems like TinyOS and NesC. Network architecture: Network scenarios, QOS parameters, design principles of WSN and the interfaces.

Unit II

Communication Protocols: Physical layer protocols - Introduction, Wireless channel and communication fundamentals, Physical layer & transceiver design considerations in WSNs. MAC layer protocols Fundamentals of (wireless) MAC protocols, Low duty cycle protocols and wakeup concepts, Contention-based protocols, Schedule-based protocols, The IEEE 802.15.4 MAC protocol, IEEE 802.11 and Bluetooth.

Unit III

Link layer protocols: Fundamentals: Tasks and requirements, Error control, Framing, Link management. Naming and addressing: Fundamentals, Address and name management in wireless sensor networks, Assignment of MAC addresses, Distributed assignment of locally unique addresses, Content-based and geographic addressing.

Unit IV

Time synchronization: Introduction to the time synchronization problem, Protocols based on sender/receiver synchronization, Protocols based on receiver/receiver synchronization. Localization and Positioning: Properties of positioning, possible approaches, Mathematical basics for the lateration problem, Single-hop localization, Positioning in multi-hop environments, Impact of anchor placement.

Unit V

Routing Protocols: The many faces of forwarding and routing, Gossiping and agent-based unicast forwarding, Energy-efficient unicast, Broadcast and multicast, Geographic routing for Mobile nodes.

Text Books:

1. Holger Karl & Andreas Willig, " Protocols And Architectures for Wireless Sensor Networks" , John Wiley, 2005.
2. Feng Zhao & Leonidas J. Guibas, “Wireless Sensor Networks- An Information Processing Approach", Elsevier, 2007.

Reference Books:

1. KazemSohraby, Daniel Minoli, &TaiebZnati, “Wireless Sensor Networks-Technology, Protocols, And Applications”, John Wiley, 2007.
2. Anna Hac, “Wireless Sensor Network Designs”, John Wiley, 2003.

Course Outcomes (COs):

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

1. Explain the hardware and software components of wireless sensor networks. (PO-2, PSO-1, 2).
2. Summarize the fundamentals of MAC protocols. (PO-2, 4, 5, PSO-1, 2)
3. Recognize the important tasks of Link layer addressing and naming schemes in WSNs (PO-1,2, 4, 5, PSO-1).
4. Identify the time synchronization problems and understand the principal design trade-offs for positioning nodes in the network. (PO-1, 2, 5, PSO-1, 2)
5. Compare the different mechanisms available for routing in the network and devise applications (PO-1,2,4, PSO-1, 2)

Software Testing

Course Code: CSE29 **Credits: 3:0:1:0**
Prerequisites: SE **Contact Hours: 42L+28P**
Course Coordinator/s: Pradeep Kumar D

Course Contents:

Unit I

A Perspective on Testing, Examples: Basic definitions, Test cases, Insights from a Venn diagram, Identifying test cases, Error and fault taxonomies, Levels of testing. Examples: Generalized pseudo code, the triangle problem, The Next Date function, The commission problem, The SATM (Simple Automatic Teller Machine) problem, The currency converter, Saturn windshield wiper.

Unit II

Boundary Value Testing, Equivalence Class Testing, Decision Table-Based Testing: Boundary value analysis, Robustness testing, Worst-case testing, Special value testing, Examples, Random testing, Equivalence classes, Equivalence test cases for the triangle problem, Next Date function, and the commission problem, Guidelines and observations. Decision tables, Test cases for the triangle problem, Next Date function, and the commission problem, Guidelines and observations. Path Testing, Data Flow Testing:DD paths, Test coverage metrics, Basis path testing, guidelines and observations, Definition-Use testing, Slice-based testing, Guidelines and observations.

Unit III

Levels of Testing, Integration Testing: Traditional view of testing levels, Alternative life-cycle models, The SATM system, separating integration and system testing. A closer look at the SATM system, Decomposition-based, call graph-based, Path-based integrations, System Testing, Interaction Testing: Threads, Basic concepts for requirements specification, Finding threads, Structural strategies and functional strategies for thread testing, SATM test threads, System testing guidelines, ASF (Atomic System Functions) testing example. Context of interaction, A taxonomy of interactions, Interaction, composition, and determinism, Client/Server Testing.

Unit IV

Process Framework: Validation and verification, Degrees of freedom, Varieties of software. Basic principles: Sensitivity, redundancy, restriction, partition, visibility, Feedback. The quality process, Planning and monitoring, Quality goals, Dependability properties, Analysis, Testing, Improving the process, Organizational factors, Fault-Based Testing, Test Execution: Overview, Assumptions in fault-based testing, Mutation analysis, Fault-based adequacy criteria, Variations on mutation analysis. Test Execution: Overview, from test

case specifications to test cases, Scaffolding, Generic versus specific scaffolding, Test oracles, Self-checks as oracles, Capture and replay.

Unit V

Planning and Monitoring the Process, Documenting Analysis and Test: Quality and process, Test and analysis strategies and plans, Risk planning, Monitoring the process, Improving the process, The quality team, Organizing documents, Test strategy document, Analysis and test plan, Test design specifications documents, Test and analysis reports.

Text Books:

1. Paul C. Jorgensen: Software Testing, A Craftsman's Approach, 3rdEdition, Auerbach Publications, 2012.
2. Mauro Pezze, Michal Young: Software Testing and Analysis –Process, Principles and Techniques, 1stEdition, WileyIndia, 2011.

Reference Books:

1. Aditya P Mathur:Foundations of Software Testing, 1stEdition,Pearson Education, 2008.
2. Srinivasan Desikan, Gopalaswamy Ramesh: Software testing Principles and Practices, 2ndEdition, Pearson Education, 2007.

Course Outcomes (COs):

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

1. Identify Test cases, Error and fault taxonomies, Levels of testing. (PO-1, 2, 12, PSO-2, 3)
2. Classify different types of testing (Boundary Value Testing, Equivalence Class Testing and Decision Table-Based Testing). (PO-1, 2, 5, 12, PSO-2, 3)
3. Recognize Alternative life - cycle models, recognize Basic concepts for requirements specification, assess context of interaction. (PO-1, 2, 3, 5, 12, PSO-2, 3)
4. Recognize approaches for Test Execution: from test case specifications to test cases, Scaffolding, Generic versus specific scaffolding. (PO-1, 2, 5, 12, PSO-2, 3)
5. Identify and plan strategies to test design specifications document. (PO-1, 2, 3, 5, 12, PSO-2, 3)

List of Lab Experiments:

1. To study and understand the need for software testing and the principles of software testing.
2. To study and learn major concepts of the testing methodologies.
3. To know different approaches to Testing, types of testing, test plan.
4. To understand black box testing and white box testing and other special type of testing.
5. To learn to create test scenarios, test cases for test data management.
6. To Study of a testing tool (e.g. Win runner).
7. To Study the usage of any web testing tool for testing (e.g. Selenium).
8. To Study the bug tracking tool for defect logging, tacking and management (e.g. Bugzilla, bugbit).
9. To Study how to use test management tool (e.g. Test Director).
10. To Study an open source-testing tool (e.g. Test Link).

Elective (Industry Collaboration)

Course Code: CSE30 **Credits: ---**
Prerequisites: Nil **Contact Hours: ---**
Course Coordinator/s: -----

- **The department will offer an Elective Course in collaboration with an Industry for those students who do not pursue their Internship in any Industry.**

- **The course to be offered will be decided by the committee in the department as per the need.**

Service Oriented Architecture and Web Services

Course Code: CSE31 **Credits: 3:0:1:0**
Prerequisites: IT **Contact Hours: 42L+28P**
Course Coordinator/s: Dr. Diwakar Harekal

Course Contents:

Unit I

Introduction to SOA, Evolution of SOA: Fundamental SOA; Common Characteristics of contemporary SOA; Common tangible benefits of SOA; An SOA timeline (from XML to Web services to SOA); The continuing evolution of SOA (Standards organizations and Contributing vendors); The roots of SOA (comparing SOA to Past architectures). Web Services and Primitive SOA: The Web services framework; services (as Web services); Service descriptions (with WSDL); Messaging (with SOAP).

Unit II

Web Services and Contemporary SOA: Message exchange patterns; Service activity; Coordination; Atomic Transactions; Business activities; Orchestration; Choreography. Addressing; Reliable messaging; Correlation; Polices; Metadata exchange; Security; Notification and eventing.
Web Services and Contemporary SOA: Message exchange patterns; Service activity; Coordination; Atomic Transactions; Business activities; Orchestration; Choreography. Addressing; Reliable messaging; Correlation; Polices; Metadata exchange; Security; Notification and eventing.

Unit III

Principles of Service – Orientation: Services-orientation and the enterprise; Anatomy of a service-oriented architecture; Common Principles of Service-orientation; How service orientation principles inter relate; Service-orientation and object-orientation; Native Web service support for service- orientation principles.

Unit IV

Service Layers: Service-orientation and contemporary SOA; Service layer abstraction; Application service layer, Business service layer, Orchestration service layer; Agnostic services; Service layer configuration scenarios. Business Process Design: WS-BPEL language basics; WS-Coordination overview; Service-oriented business process design; WS-addressing language basics; WS-Reliable Messaging language basics.

Unit V

SOA Platforms: SOA platform basics; SOA support in J2EE; SOA support in .NET; Integration considerations.

Text Book:

1. Thomas Erl: Service-Oriented Architecture – Concepts, Technology, and Design, Pearson Education, 2005.

Reference Book:

1. Eric Newcomer, Greg Lomow: Understanding SOA with Web Services, Pearson education, 2005.

Course Outcomes (COs):

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

1. Summarize Web Service and Service oriented Architecture. (PO-1, 2, 3, 4, PSO-1)
2. Illustrate the principles of contemporary SOA and web service. (PO-2, 4, 5, PSO-1, 2)
3. Appraise the principles the layers of Service Oriented Architecture. (PO-2, 3, PSO-1, 2)
4. Infer the service oriented principles (PO-2, 3, 4, PSO-1, 2)
5. Categorize SOA support in J2EE and SOA support in .NET focusing on platform overview. (PO-2, 3, 6, 12, PSO-1, 2)

List of Lab Exercises:

1. To develop a web service program which can persists the records of a student in the exam table.
2. It makes use of SOAP Request and SOAP Response.
3. To invoke EJB components as web services.
4. To create a web services in .NET.
5. To invoke j2ee web services from .net clients.
6. To create components using .NET client.
7. To access .net web services from J2EE client.
8. Develop a Service Orchestration Engine (workflow) using WS-BPEL and implement service composition (Study Experiment).

Project Work

Course Code: CSP **Credits: 0:0:16:0**
Prerequisites: Nil **Contact Hours: -**
Course Coordinator/s: Dr. Monica R Mundada, Sowmya B J, Dr. S Rajarajeswari

Course Contents:

As a part of project, all the eligible final year students must carry out the following activities:

1. Students should form a group to carry out their project. The minimum group size is 2 and maximum group size is 4.
2. The groups will be attached to one Internal Guide (and Co-guide if necessary) by the Department.
3. Students can carry out their project in-house or in a reputed organization (to be approved by Internal Guide and HOD).
4. Identify the problem statement based on the current state of Art and trends in the area of interest.
5. Based on the survey, identify the project requirements and do feasibility study.
6. Identify and draw a system level architecture by showing subsystems and their input/output need.
7. Implement the programs using step by step for each module.
8. Integrate and examine the implementation and test the project scope and the requirements.
9. Prepare Project document and the demonstrating their work.
10. The evaluation is based on presentation and report.
    - The evaluation will be done by the internal guide and a co-examiner twice during the semester.
        - Mid-semester evaluation: Students must do a group presentation and produce documents of system requirements, and system design (during 6th week)
        - Final Evaluation: At the End of the semester students must do a group presentation, demonstrate the project work and submit the complete report. (during 13th week)

Course Outcomes (COs):

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

1. Review the current state of Art and trends in their area of interest and identify a suitable problem in their chosen subject domain with justification. (PO-1,2,3,4,5,6, 9, 10, 11,12, PSO-2,3)
2. Survey the available research literature/documents for the tools and techniques to be used. (PO-1, 2, 5, 8, 9, 10, 11, 12, PSO-2,3)
3. Examine the functional, non-functional, and performance requirements of their chosen problem definition. (PO-1,2,4, 9, 10, 11, 12, PSO-2,3)
4. Design system architecture and different components and develop all the system components using appropriate tools and techniques. (PO-1, 2, 3, 4, 5, PSO-2,3)
5. Work effectively in a team and use good project management practices and defend the project work as a team(PO-5, 8, 9, 10, 11, 12, PSO-2,3)

Rubrics for Project Evaluation:

The rubrics for project evaluation is provided in separate project work book maintained for each project team.

Extra-curricular and Co-curricular Activities

Course Code: EAC **Credits: 0:0:0:2**
Prerequisites: Nil **Contact Hours: -**
Course Coordinator/s: --

Details for Extra-Curricular and Co-Curricular Activities

In the Uniform Teaching Scheme for UG from 2015-16 batch, two credits are allocated to Extra-Curricular and Co-Curricular Activities (EAC).
The student is made aware of the credits allotted to EAC at the beginning of the First semester by the respective Department Co-ordinator/Proctor. The evaluation procedure is as follows:

- Each student need to submit the evidence for the claims for the relevant categories mentioned in the table for evaluation
- If any student has a significant contribution in any category other than the above mentioned need to submit the report with proof
- The evaluation is done when the student is in 8^{th} semester
- The evaluation rubrics have to be made known to the students (mentioned in the table 3)
- The department have to clearly specify/justify the rubrics for evaluation
- The evaluation team involves Proctor/HOD/Committee(faculty nominated by HOD)

Course Outcomes (COs):

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

1. Demonstrate their talents and gain confidence to participate in extracurricular activities in future (PO-6,7,9,10,12)
2. Improve their self-thinking, self- understanding to promote their individual growth and balance between academics and outside commitments (PO-6,8,9,12)
3. Demonstrate enhanced communication and public speaking skills, organizational skills, leadership skills and work in multidisciplinary teams with positive attitude (PO-6,7,9,10,11,12, PSO-3)

Evaluation of Extra-Curricular and Co-Curricular Activities
1: Extra-Curricular Activity (50 Marks)

S. No.	Name of the Activity	MSRIT Level	State Level	National Level	International Level
1.	Participation/Won Prizes in any event of the Sports Festivals/Cultural festivals				
2.	Representing the Institute in any sports event/cultural event				
3.	Invitation to participate as referee/expert advisor/official/Captain in any sports event/cultural event				
4.	Won scholarships/grants for any sports event/cultural event				
5.	Team member/individual representing the State/Country in any sports event/cultural event				
6.	Participation/Won awards/received grants through NCC/NSS/NGO activities				

2: Co-Curricular Activity (50 Marks)

S. No.	Name of the Activity	MSRIT Level	State Level	National Level	International Level
1.	Participation/Won Prizes in any event of the Technical Festivals				
2.	Delivering lectures/handling hands on sessions on technical topics in any technical event				
3.	Invitation to participate as a representative from the Institute to any technical event				
4.	Invitation to participate as referee/Expert/Judge to any technical event				
5.	Won merit scholarships/Consultancy/award with monetary benefits/grants for any technical event				
6.	Entrepreneur activity like setting up a company in association with EDC cell of MSRIT				
7.	Team member/individual representing the State/Country in any technical event				
8.	Participation/Presented Research Papers/Won Prizes in Conference/Workshop/Symposium/Seminar				
9.	Published Research Papers in refereed Journals				
10.	Under-going In-Plant training				
11.	Helping the faculty in Consultancy, Testing and R&D Projects				
12.	Prototype Preparation as a teaching aid				
13.	Conduction of Career Guidance Programmes				

3: Rubrics for the assessment of Extra-Curricular and Co-Curricular Activities

Assessment Criteria	MSRIT Level	State Level	National Level	International Level
1. Student has won a prize in the event	**38-45**	**38-45**	**45-50**	**45-50**
2. Student has progressed to the last level of the event	**25-38**	**38-45**	**45-50**	**45-50**
3. Student has participated in the event	**0-25**	**25-38**	**38-45**	**45-50**

Internship

Course Code: CSIN **Credits: 0:0:0:4**
Prerequisites: Nil **Contact Hours: -**
Course Coordinator/s: --

Rubrics for assessment of Internship

Deliverables for Student Performance in Internship:

Internship Title: _____

Company Name:
_____

Name of Student:
_____

Name of Supervisor at Company:
_____

Name of Supervisor at College:
_____

Each supervisor must fill a rubric for each student:

Tools and new Technology Learnt	**Basic (0-4 Pts)** Few sources at the Industry, aware of quality of resources and relevance to tools and Techniques at hand	**Good (5-7 Pts)** Multiple sources of high quality, good judgment of the information, identification of gaps in knowledge at the Industry and Academics.	**Very Good (10 Pts)** Multiple sources of high quality, well researched and analyzed, continuous efforts at acquiring Information. Identification of the application of the tools and Technology learnt to the present market.	**Total**
Relevance of the topic chosen to the current market	Fairly Relevant	Moderately Relevant	Highly Relevant	

- Rubrics for the Report Writing, Demo and Presentation are maintained in a separate work book

Institutional Elective
Python Application Programming

Course Code: CSOE01 **Credits: 3:0:0:1**
Prerequisites: Nil **Contact Hours: 42L**
Course Coordinator/s: Srinidhi H

Course Contents:

Unit I

The way of the program: The Python programming language, What is a program?, What is debugging?, Syntax errors, Runtime errors, Semantic errors, Experimental debugging.
Variables, Expressions and Statements: Values and data types, Variables, Variable names and keywords, Statements, Evaluating expressions, Operators and operands, Type converter functions, Order of operations, Operations on strings, Input, Composition, The modulus operator. **Iteration:** Assignment, Updating variables, The for loop, The while statement, The Collatz 3n + 1 sequence, Tables, Two-dimensional tables, Paired Data, Nested Loops for Nested Data. **Strings:** Working with strings as single things, Working with the parts of a string, Length, Traversal and the for loop, Slices, String comparison, Strings are immutable, The in and not in operators, A find function, Looping and counting, Optional parameters, The built-in find method, The split method, Cleaning up your strings, The string format method.
Self-Study: Conditional Statements.

Unit II

Tuples: Tuples are used for grouping data, Tuple assignment, Tuples as return values, Composability of Data Structures.
Lists: List values, Accessing elements, List length, List membership, List operations, List slices, Lists are mutable, List deletion, Objects and references, Aliasing, Cloning lists, Lists and for loops, List parameters, List methods, Pure functions and modifiers, Functions that produce lists, Strings and lists, list and range, Nested lists, Matrices.
Dictionaries: Dictionary operations, dictionary methods, aliasing and copying.
Functions: Functions with arguments and return values.
Self-Study: Lambda Functions.

Unit III

Modules: Random numbers, The time module, The math module, Creating your own modules, Namespaces, Scope and lookup rules, Attributes and the dot operator.
Files: About files, Writing our first file, Reading a file line-at-a-time, Turning a file into a list of lines, Reading the whole file at once, Working with binary files, Directories, fetching something from the web.
Self-Study: Algorithms: Linear search, Binary search, merging two sorted lists.

Unit IV

Object oriented programming: Classes And Objects — The Basics, Attributes, Adding methods to our class, Instances as arguments and parameters, Converting an instance to a string, Instances as return values, Objects are mutable, Sameness, Copying.
Exceptions: Catching exceptions, Raising our own exceptions, The finally clause of the try statement
Inheritance: Polymorphism, Generalization, Pure functions.
Self-Study: Operator Overloading.

Unit V

GUI: Creating Graphical User Interfaces, Using Module Tkinter, Building a Basic GUI, Models, Views, and Controllers, Customizing the Visual Style, Few More Widgets.
Databases: Overview, Creating and Populating, Retrieving Data, Updating and Deleting, Using NULL for Missing Data, Using Joins to Combine Tables, Keys and Constraints, Advanced Features.
Self-Study: Developing GUI Application.

Text Books:

1. Downey, A., Elkner, J., & Meyers, C. (2002). How to think like a computer scientist: learning with python. Green Tea Press, Wellesley, Massachusetts.
2. Campbell, J., Gries, P., Montojo, J., & Wilson, G. (2013). Practical programming: an introduction to computer science using Python. Pragmatic Bookshelf, Second Edition.

Course Outcomes (COs):

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

1. Design and implement Python programs utilizing arithmetic expressions, repetition, file Input and Output. (PO-1,2,3,5,12)
2. Define and demonstrate the use of the built-in data structures in Python. (PO-1,2,3,5,12)
3. Employ control structures, functions, and arrays to create Python programs. (PO-1,2,3,5,12)
4. Understand the concepts of object-oriented programming as used in Python. (PO-1,2,3,5,12)
5. Define and demonstrate the use of GUI and databases using Python. (PO-1,2,3,5,12)

Institutional Elective
Mobile Application Development

Course Code: CSOE02 **Credits: 3:0:0:1**
Prerequisites: OOPs and Java **Contact Hours: 42L**
Course Coordinator/s: Pramod C Sunagar

Course Contents:

Unit I

Android Programming, Activities, Fragments, and Intents: What Is Android? Obtaining the Required Tools, Creating Your First Android Application, Anatomy of an Android Application. Using the Android Emulator, **Activities, Fragments, and Intents:** Understanding Activities, Linking Activities Using Intents.
Self-Study: Fragments, Calling Built-In Applications Using Intents, Displaying Notifications.

Unit II

User Interface, Designing User Interface with Views: Understanding the Components of a Screen, Adapting to Display Orientation, Managing Changes to Screen Orientation, Utilizing the Action Bar, Creating the User Interface Programmatically, Listening for UI Notifications, **Designing User Interface with Views**: Using Basic Views, Using Picker Views, Using List Views to Display Long Lists, Understanding Specialized Fragments, Menu.
Self-Study: Multimedia widgets and its working.

Unit III

Data Persistence, Content Providers: Saving and Loading User Preferences, Persisting Data to Files, Creating and Using Databases. **Content Providers**: Sharing Data in Android, Using a Content Provider, Creating Your Own Content Providers, Using the Content Provider.
Self-Study: Shared Preferences.

Unit IV

Messaging, Android Services:
SMS Messaging, Sending E-mail Services in Android. **Android Services:** Types of Services: Local Service, Remote Service, Intent Service. Broadcast Receivers, Types of Broadcasts, Creating a Broadcast Receivers.
Self-Study: Notifications and other classes.

Unit V

Location Based Services, Publishing Android Applications:
Using Location-Based Services, Configuring the Emulator to Test Location-Based Services, Updating Locations in Emulator Location Providers, Selecting a Location Provider, Finding Your Location, Using Proximity Alerts, Using the Geocoder, Creating Map-Based Activities, Mapping Earthquakes Example Using Background Threads.
Self-Study: Preparing for Publishing, Deploying APK Files.

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. "Beginning Android Application development", Wei-Meng Lee, Wiley Publishing,
2. Professional Android 2 Application Development by Reto Meier, Wiley Publishing, 2010

Reference Books:

1. Android 4.2 App Development Essentials, by Neil Smyth - Techotopia , 2013
2. The Busy Coder's Guide to Android Development, by Mark L. Murphy – CommonsWare , 2009

Course Outcomes (COs):

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

1. Understand the android OS and fundamental concepts in android programming. (PO-1,2,3,5,12)
2. Demonstrate various components, layouts and views in creating android applications. (PO-1,2,3,5,12)
3. Design applications to save, or store, data using SQLite and Content Providers. (PO-1,2,3,5,12)
4. Demonstrate the working of long running tasks in the background using Services. (PO-1,2,3,5,12)
5. Demonstrates how to write applications using location based services and to publish the android applications. (PO-1,2,3,5,12)

Institutional Elective
Internet of Things (IoT)

Course Code: CSOE03 **Credits: 3:0:0:1**
Prerequisites: Nil **Contact Hours: 42L**
Course Coordinator/s: Hanumantha Raju R

Course Contents:

Unit I

Introduction to IoT: What is IoT? , IOT terms and Basic Definitions, Disambiguation of IoT vs IoE vs M2M vs Others, Characteristics of IoT.
Wireless Sensor Networks: Potential Applications, WSN System Architecture, WSN Network Topologies, Components of a WSN Node.
Architecture of IoT systems: Things in IoT, Applications of IoT and IoT Reference model.
Self-Study: IoT Ecosystem, Enabling Technologies in IoT, Marketplace and Vision of IoT.

Unit II

Hardware aspects of IoT: Sensors and Actuators:
Introduction to Sensors: Workflow of a Sensor in a typical system, Classification of Sensors, Sampling DAC and ADC conversion.
Introduction to Actuators: Workflow of an Actuator in a typical system, Classification of Actuators.
Self-Study: Types of Sensors, Interfacing concepts to embedded systems.

Unit III

Communications and networking aspects of IoT:
High bandwidth networking: Ethernet, gigabit Ethernet, Ethernet topologies like bridge and switches, Passive optical fiber network and topologies, WiFi and WiMax. WiFi routers, radius servers, Wireless security with WPA-2, LEAP, enterprise WPA networks
Low Bandwidth Wireless Networks: FSK, LoRa modulation basics, LoRaWAN basics.
Peripherals networking: Basics of I2C, SPI, RS232, RS485 and CAN bus, Comparisons and use cases of I2C, SPI, RS232, RS485and CAN bus.
Self-Study: Introduction to BLE 5 and industrial Wireless sensor networks, Security in low bandwidth wireless networks, Security in peripheral networks.

Unit IV

Software and middleware aspects of IoT:-
Middleware: Components of Middleware, Types of Databases, Micro services and API's.
IP Communication protocols: HTTP, AMQP, MQTT and STOMP.
Self-Study: Protocol definitions, use cases and differences.

Unit V

IoT Platform Design Methodology and Domain Specific IoT.
Self-Study: Futuristic view of IoT, problems pertaining to implementation like scaling, feasibility and management.

Text Books:

1. Srinivasa K G,Siddesh G.M and Hanumantha Raju R "Internet of Things", CENGAGE Leaning India, 2017. (ISBN:978-93-868-5895-5).

References:

1. Vijay Madisetti and ArshdeepBahga, "Internet of Things (A Hands-on-Approach)", 1stEdition, VPT, 2014. (ISBN: 978-8173719547).
2. Designing the Internet of Things by Adrian McEwenSmart Cities, Software above the level of a single device,Ebooks on IoT by O'Reilly
3. Sentilo middleware http://www.sentilo.io/xwiki/bin/view/Sentilo.About.Product/Whatis
4. Mosquitto broker https://mosquitto.org/
5. Getting started with raspberry pi https://www.raspberrypi.org/resources/learn/
6. Arduino basics https://www.arduino.cc/en/Tutorial/HomePage
7. Wired peripheral protocols http://www.comm.pub.ro/dicm/C7_Serial_Bus.pdf
8. OneM2m http://www.indiaeu-ictstandards.in/wp-content/uploads/2017/04/oneM2M-for-smart-city-TSDSI-presentation-April-21st-2017-Omar-Elloumi.pdf
9. LoRa Modulation http://www.semtech.com/images/datasheet/an1200.22.pdf

Course Outcomes (COs):

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

1. Understand the basics of IoT. (PO-1,2,3,5,12)
2. Demonstrate various components, layouts and views in creating IoT applications. (PO-1,2,3,5,12)
3. Design applications using sensors and actuators. (PO-1,2,3,5,12)
4. Demonstrate the working of long running tasks in the background using IoT. (PO-1,2,3,5,12)
5. Demonstrate how to write applications for smart world. (PO-1,2,3,5,12)