CURRICULUM
for the Academic year 2018 – 2019

DEPARTMENT OF COMPUTER SCIENCE
AND ENGINEERING

V & VI SEMESTER B.E

RAMAIAH INSTITUTE OF TECHNOLOGY
(Autonomous Institute, Affiliated to VTU)
BANGALORE – 54
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 2016 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 150 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 60th rank in 2018 among the top 100 engineering colleges across India.
About the Department

<table>
<thead>
<tr>
<th>Year of Establishment</th>
<th>1984</th>
</tr>
</thead>
<tbody>
<tr>
<td>Names of the Programmes offered</td>
<td></td>
</tr>
<tr>
<td>1. UG: B.E. in Computer Science and Engineering</td>
<td></td>
</tr>
<tr>
<td>2. PG: M.Tech. in Computer Science and Engineering</td>
<td></td>
</tr>
<tr>
<td>3. PG: M.Tech. in Computer Networks and Engineering</td>
<td></td>
</tr>
<tr>
<td>4. Ph.D</td>
<td></td>
</tr>
<tr>
<td>5. M.Sc(Engg.) by Research</td>
<td></td>
</tr>
</tbody>
</table>

The Department of Computer Science and Engineering (CSE) has eminent emeritus professors, 10 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, IITBombayX, 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

- Imparting quality technical education by nurturing a conducive learning environment through continuous improvement and customization
- Establishing research clusters in emerging areas in collaboration with globally reputed organizations
- 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 stakeholders concerned

VISION OF THE DEPARTMENT

To build a strong learning and research environment in the field of Computer Science and Engineering that responds to the challenges of 21st century.

MISSION OF THE DEPARTMENT

- To produce computer science graduates who, trained in design and implementation of computational systems through competitive curriculum and research in collaboration with industry and other organizations
- To educate students in technology competencies by providing professionally committed faculty and staff
- 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
PROGRAM 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

PROGRAM 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.

PROGRAM 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**

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

**SCHEME OF TEACHING**

**V SEMESTER**

<table>
<thead>
<tr>
<th>SI. No</th>
<th>Course Code</th>
<th>Course Name</th>
<th>Category</th>
<th>Credits</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>CS51</td>
<td>Operating Systems</td>
<td>PC-C</td>
<td>3</td>
<td>42</td>
</tr>
<tr>
<td>2.</td>
<td>CS52</td>
<td>Database Systems</td>
<td>PC-C</td>
<td>3</td>
<td>42+28</td>
</tr>
<tr>
<td>3.</td>
<td>CS53</td>
<td>Computer Networks</td>
<td>PC-C</td>
<td>4</td>
<td>56</td>
</tr>
<tr>
<td>4.</td>
<td>CS54</td>
<td>Java Programming</td>
<td>PC-C</td>
<td>3</td>
<td>42</td>
</tr>
<tr>
<td>5.</td>
<td>CS55</td>
<td>Intellectual Property Rights</td>
<td>HSS</td>
<td>2</td>
<td>28</td>
</tr>
<tr>
<td>6.</td>
<td>CSExx</td>
<td>--</td>
<td>PC-E</td>
<td>*</td>
<td>56</td>
</tr>
<tr>
<td>7.</td>
<td>CSL57</td>
<td>Java Programming Laboratory</td>
<td>PC-C</td>
<td>0</td>
<td>28</td>
</tr>
<tr>
<td>8.</td>
<td>CSL58</td>
<td>Database Systems Laboratory</td>
<td>PC-C</td>
<td>0</td>
<td>28</td>
</tr>
<tr>
<td>9.</td>
<td>CSL59</td>
<td>Computer Networks Laboratory</td>
<td>PC-C</td>
<td>0</td>
<td>28</td>
</tr>
<tr>
<td></td>
<td><strong>Total</strong></td>
<td></td>
<td></td>
<td>1</td>
<td>25</td>
</tr>
</tbody>
</table>

---
# Professional Elective List for V Semester

<table>
<thead>
<tr>
<th>Sl. No</th>
<th>Course code</th>
<th>Course Name</th>
<th>Credits</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>CSE01</td>
<td>Data Mining (3:0:1)</td>
<td>4</td>
<td>42+28</td>
</tr>
<tr>
<td>2.</td>
<td>CSE02</td>
<td>Artificial Intelligence (3:0:1)</td>
<td>4</td>
<td>42</td>
</tr>
<tr>
<td>3.</td>
<td>CSE03</td>
<td>Operation Research (4:0:0)</td>
<td>4</td>
<td>42+28</td>
</tr>
<tr>
<td>4.</td>
<td>CSE04</td>
<td>Advanced Algorithms (3:0:1:0)</td>
<td>4</td>
<td>42+28</td>
</tr>
<tr>
<td>5.</td>
<td>CSE05</td>
<td>System Simulation (4:0:0)</td>
<td>4</td>
<td>42+28</td>
</tr>
<tr>
<td>6.</td>
<td>CSE06</td>
<td>Object Oriented Modelling and Design (3:0:1:0)</td>
<td>4</td>
<td>42+28</td>
</tr>
</tbody>
</table>

## VI SEMESTER

<table>
<thead>
<tr>
<th>Sl. No</th>
<th>Course Code</th>
<th>Course Name</th>
<th>Category</th>
<th>Credits</th>
<th>Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>CS61</td>
<td>Compiler Design</td>
<td>PC-C 3 1 0 0 4</td>
<td>42+28</td>
<td></td>
</tr>
<tr>
<td>2.</td>
<td>CS62</td>
<td>Software Engineering</td>
<td>PC-C 3 0 0 1 4</td>
<td>42</td>
<td></td>
</tr>
<tr>
<td>3.</td>
<td>CS63</td>
<td>Unix System Programming and Web Technologies</td>
<td>PC-C 3 1 0 0 4</td>
<td>42+28</td>
<td></td>
</tr>
<tr>
<td>4.</td>
<td>CS64</td>
<td>Mini Project-Themes: Software Development for Portable Device/Rich InternetApplications/Embedded Systems</td>
<td>PW 0 0 4 2 6</td>
<td>56</td>
<td></td>
</tr>
<tr>
<td>5.</td>
<td>CSEXx</td>
<td>--</td>
<td>PC-E * * * * 4</td>
<td>56</td>
<td></td>
</tr>
<tr>
<td>6.</td>
<td>CSL66</td>
<td>Unix System Programming &amp; Compiler Design Laboratory</td>
<td>PC-C 0 0 1 0 1</td>
<td>28</td>
<td></td>
</tr>
<tr>
<td>7.</td>
<td>CSL67</td>
<td>IoT / Embedded Systems Laboratory</td>
<td>PC-C 0 0 1 0 1</td>
<td>28</td>
<td></td>
</tr>
<tr>
<td>8.</td>
<td>CSL68</td>
<td>Web Technologies Laboratory</td>
<td>PC-C 0 0 1 0 1</td>
<td>28</td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td></td>
<td></td>
<td>13 2 7 3 25</td>
<td>--</td>
</tr>
<tr>
<td>SL. No</td>
<td>Course Code</td>
<td>Course Name</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
<td>--------------------------------------------------</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.</td>
<td>CSE07</td>
<td>Mobile Computing (3:0:1:0)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.</td>
<td>CSE08</td>
<td>Computer Graphics and Visualization (3:0:1:0)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3.</td>
<td>CSE09</td>
<td>Software Defined Networks (3:0:0:1)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.</td>
<td>CSE10</td>
<td>Soft Computing (4:0:0:0)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5.</td>
<td>CSE11</td>
<td>Machine Learning (3:0:1:0)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6.</td>
<td>CSE12</td>
<td>Natural Language Processing (3:0:1:0)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7.</td>
<td>CSE13</td>
<td>Information Retrieval (3:0:1:0)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

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

Course Code: CS51  
Prerequisites: Nil  
Course Coordinator/s: Vandana Sardar

Course Contents:

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

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

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

Unit IV
Self-Study: Dining Philosophers, Deadlock detection & Recovery from deadlock.
Unit V

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

**Self-Study** t:-FSCK and journaling: Introduction, A detailed example & A simple disk drive.

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

**Text Books:**
1. Operating systems: Three easy pieces, by Remzi H. Arpaci-Dusseau and Andrea C. Arpaci-Dusseau

**Reference Books:**

**Course Outcomes (COs):**

At the end of the course, the student must be able to:
1. Explain operating systems objectives and its various functions(PO-1,3, PSO-1)
2. Apply different process scheduling algorithms and measure their performance(PO-1,2,3,4, PSO-1)
3. Illustrate various memory management schemes such as paging, segmentation and swapping mechanism(PO-1,2,3,4, PSO-1)
4. Assess different concurrency control techniques to provide co-ordination among processes(PO-1,2,3,4, PSO-1)
5. Describe organization of files and directories including the merits of using fast file systems(PO-1,2,3, PSO-1)
Course Code: CS52  
Prerequisites: Nil  
Course Coordinator/s: Ganeshayya I Shidaganti

Course Contents:

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

Unit II

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

Unit IV

Unit V
Text Books:


Reference Book:


Course Outcomes (COs):

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

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

Course Code: CS53
Prerequisites: Data Communication
Course Coordinator/s: Sanjeetha R, Darshana Naik

Credits: 4:0:0:0
Contact Hours: 56

Course Contents:

Unit I


Unit III
IPv4 Addresses: Classful addressing, classless addressing, Transition from IPv4 to IPv6
Routing and Internetworking - Network-layer routing, least-cost-path algorithms, Intra domain routing protocols, Inter domain routing protocols, Congestion Control at Network Layer.

Unit IV

Unit V
Text Books:


Reference Books:


Course Outcomes (COs):

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

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

Course Code: CS54
Prerequisites: OOPS concepts
Credits: 3:0:0:1
Contacts: J Geetha, Hanumantharaju R

Course Contents:

Unit I
Self-Study: Java Buzz Words, Arrays, Strings, String Buffer.

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

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

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

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

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

Text Books:


Reference Books:


Course Outcomes (COs):

At the end of the course, the student will be able to:
1. Understand the paradigm of java programming (PO-2,3,5 PSO-2)
2. Develop java programs using multithreaded and event handling concepts. (PO-2,3,5 PSO-2)
3. Identify the different methods of creating querying the database. (PO-2,3,5 PSO-2)
4. Design and develop the web applications using Servlets. (PO-2,3,5 PSO-2)
5. Design the dynamic web pages using JSP. (PO-2,3,5 PSO-2)
Intellectual Property Rights

Course Code: CS55
Prerequisites: Nil
Course Coordinator/s: Dr. T.N.R. Kumar
Course Contents:

Unit I


Unit II

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

Unit III

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

Unit IV

Copy Right: Meaning and characteristics of copy right, Indian copy right law, requirement of copy right, Illustrations copy right in literary work, Musical work, Artistic work, work of architecture, Cinematograph film, sound recording. Author and Ownership of copy right: Ownership of copy right, Contract of service, Contract for service, rights conferred by copy right, terms of copy right, license of copy right. Infringement of copy right: Acts which constitute infringement, general principle, direct and indirect evidence of copying, Acts not constituting infringements, Infringements in literary, dramatic and musical works, Remedies against infringement of copy right, Case studies.

Unit V

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

1. Dr. T Ramakrishna: Basic principles and acquisition of Intellectual Property Rights, CIPRA, NSLIU -2005.

References Books:


Course Outcomes (COs):

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

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

Course Code: CSE01
Prerequisites: Nil
Course Coordinator/s: Dr. Seema S, Sowmya B J
Course Contents:

Unit I
Association Rule Mining Mining Frequent Patterns – Apriori Algorithm Description.
Self-Study: Normalization.

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

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

Unit IV
Self-Study: Data Mining Applications.

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

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


Reference Books:


Course Outcomes (COs):

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

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

Course Code: CSE02

Credits: 3:0:0:1
Contact Hours: 42

Prerequisites: Knowledge of any advanced programming language, Algorithms and Data structures, Elementary Discrete Mathematics or similar.
Course Coordinator/s: Dr. S. Rajarajeswari and Dr. Annapurna P Patil

Course Contents:

Unit I

Self-Study: On-line search agents and unknown environments.

Unit II

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

Unit III


Unit IV

Unit V


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

Text Books:


Reference Books:

3. http://nptel.ac.in

Course Outcomes (COs):

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

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

Course Code: CSE03
Prerequisites: Nil
Course Coordinator/s: Dr. Jagdish S Kallimani

Course Contents:

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

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

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

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

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

Text Book:

Reference Books:


Course Outcomes (COs):

At the end of the course the student will be able to:
1. Explain the concepts of LPP and formulations. (PO-1,2, PSO-2)
2. Construct problems under simplex methods and its types. (PO-1,2, PSO-2)
3. Identify problems under duality and justify them. (PO-3,5,7, PSO-2)
4. Identify assignment and transportation problems. (PO-4, PSO-2)
5. Explain game theory and decision making problems. (PO-4,11, PSO-2)
Advanced Algorithms

Course Code: CSE04
Prerequisites: Algorithms
Course Coordinator/s: Dr. Jagadish S Kallimani

Credits: 3:0:1:0
Contact Hours: 42+28

Course Contents:

Unit I

Unit II

Unit III

Unit IV

Unit V

Text Books:

Reference Books:
Course Outcomes (COs):

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

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

List of Lab Exercises:

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

1. Graph Algorithms.
2. B-Trees.
3. RB Trees.
4. Hashing.
5. Heaps.
8. Algorithmic Puzzles and Others.
System Simulation

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

Course Contents:

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

Unit II

Unit III

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

Unit V

Text Book:

Reference Books:


Course Outcomes (COs):

At the end of the course, the student will be able to:
1. Identify modelling system and types of simulation tools. (PO-1,2 PSO-2)
2. Understand the concepts of scheduling /Queueing system using simulation software. (PO-1,2, PSO-2)
3. Test and Analyze random function generation through various transform techniques. (PO-1,2, PSO-2)
4. Analyze the data collection process. (PO-1,2, PSO-2)
5. Interpret the stochastic nature of output data. (PO-1, 2, PSO-2)
Object Oriented Modeling and Design

Course Code: CSE06  
Prerequisites: Nil  
Course Coordinator/s: Meeradevi K

Course Contents:

**Unit I**

**Unit II**

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

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

**Unit V**

Text Books:

3. http://sourcemaking.com/design_patterns (Unit : 2)
Reference Books:


Course Outcomes (COs):

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

1. Design projects using OO concepts and UML diagrams. (PO-1,2,3,4, 11,12, PSO-2,3)
2. Use the UML analysis and design diagrams. (PO-2,3,5,11, PSO-2,3)
3. Apply appropriate design patterns. (PO-4,5,6,7,8,PSO-2,3)
4. Create code from design and implement using oo concepts. (PO-2,3,5, 9, PSO-2,3)
5. Compare and contrast various testing techniques. (PO-4,5,9,10,11, PSO-2,PSO-3)

List of Lab Exercises:

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

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

Course Contents:

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

Text Books:


Reference Books:


Course Outcomes (COs):

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

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

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

Reference Books:


Course Outcomes (COs):

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

1. Recognize the Core MongoDB Operations(PO-2,5,PSO-1,2)
2. Design an Oracle DB Application Using SQL Queries(PO-1,3,4,5,PSO-1,2)
3. Develop a Real time database Application Using IDE(PO-1,3,4,5,PSO-1,2)
Computer Networks Laboratory

Course Code: CSL59                        Credits: 0:0:1:0
Prerequisites: Data Communication       Contact Hours: 28
Course Coordinator/s: Sanjeetha R, Darshana Naik

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

PART – A

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

PART-B

The following experiments shall be conducted using either NS-2/NS3/OMNET++ or any other suitable simulator.
1. Simulate a three nodes point-to-point network with duplex links between them. Set the queue size vary the bandwidth and find the number of packets dropped.
2. Simulate a four node point-to-point network, and connect the links as follows: n0-n2, n1-n2 and n2-n3. Apply TCP agent between n0-n3 and UDP agent between n1-n3. Apply relevant applications over TCP and UDP agents by changing the parameters and determine the number of packets sent by TCP/UDP.
3. Simulate simple Extended Service Set with transmitting nodes in wireless LAN and determine the performance with respect to transmission of packets.
4. Simulate a wireless network, generate traffic and analyze its performance.
5. Simulate a transmission of ping message over a network topology consisting of 6 nodes and find the number of packets dropped due to congestion.

Reference Books:


Course Outcomes (COs):

At the end of the course the student will be able to:
1. Illustrate networking concepts using programming languages like C/C++/Java/Python. (PO-1, 2, 3, 4, 5, PSO-1)
2. Use packet sniffing tools like Wireshark to intercept & analyze the packets at different network layers. (PO-1, 2, 3, 4, 5, PSO-1)
3. Use simulators like NS2/NS3. (PO-1, 2, 3, 4, 5, PSO-1)
Compiler Design

Course Code: CS61
Prerequisites: Nil
Course Coordinator/s: A Parkavi

Course Contents:

Unit I

Unit II

Unit III

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

Unit V

Text Book:


Reference Books:

Course Outcomes (COs):

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

1. Construct lexical analyzer to recognize inputs using patterns. (PO-1, 2, 3, 4, 5, PSO-1)
2. Devise different types of syntax analyzers using grammars. (PO-1, 2, 3, 4, 5, PSO-1, 2)
3. Illustrate syntax-directed translation schemes for grammars. (PO-1, 2, 3, 4, 5, PSO-1)
4. Formulate intermediate code generators for programming statements. (PO-1, 2, 3, 4, 5, PSO-1)
5. Develop assembly language code for the given optimized intermediate codes. (PO-1, 2, 3, 4, 5, PSO-1, 2)
Software Engineering

Course Code: CS62
Prerequisites: Nil
Course Coordinator/s: Dr. Annapurna P Patil

Credits: 3:0:0:1
Contact Hours: 42

Course Contents:

**Unit I**
**Self-Study**- Activity based learning of Process models.

**Unit II**
**Planning a Software Project:** Effort Estimation, Project Schedule & Staffing, Quality Planning, Risk Management Planning, Project Monitoring Plan.
**Self-Study**- Case study for planning.

**Unit III**
**Coding:** Programming Principles & Guidelines, Incrementally Developing Code, Managing Evolving Code.
**Self-Study**- Design activities.

**Unit IV**
**Self-Study**- study of testing tools

**Unit V**
**Self-Study**- role play for agile

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


Reference Books:


Course Outcomes (COs):
At the end of the course, the students should be able to:

1. Recall the principles and techniques of Software Engineering. (PO-2,3,9,10,11, PSO-2,3)
2. Appraise the activities in project management, requirement engineering process and the different types of system models. (PO-2,3,9,10,11, PSO-2,3)
3. Illustrate the knowledge of design engineering in software development. (PO-2,3,9,10,11, PSO-2,3)
4. Compare different testing methods and tools. (PO-2,3,9,10,11, PSO-2,3)
5. Describe the need for software engineering practices for Web Engineering. (PO2,3, PSO-2,3)
UNIX System Programming and Web Technologies

Course Code: CS63
Prerequisites: Nil
Course Coordinator/s: Aparna R

Credits: 3:1:0:0
Contact Hours: 42+28

Course Contents:

Unit I

Unix Basics & Fundamentals of JavaScript:


Unit II

Process and JavaScript Advanced:

JavaScript Advanced: Scopes and Closures, Understand "this" and prototypes, OO concepts as applied to JS and prototypal inheritance, Understanding the meaning of asynchronous.

Unit III

Process Control and Node.js


Unit IV

Signals and Node.js with MONGODB

CRUD Operations using Node.js: Event Handling - GET & POST implementation, Use Express.js to create a REST API. Use GET, POST. Connect to NoSQL Database using Node.js, Implementation of CRUD operations.
Unit V
Introduction to Client-side JS Framework – Basics of Angular 4.0:
Introduction to Angular 4.0, Needs & Evolution, Features-Setup and Configuration, Components and Modules–Templates, Change Detection, Directives, Data Binding, Pipes, Nested Components, Model Driven Forms or Reactive Forms.

Text Books:

Reference Books:
4. Web link for Angular4.0: https://angular.io/

Course Outcomes (COs):
At the end of the course, the student will be able to:
1. Describe the functions available for file I/O and changing the properties of the file in Unix OS. (PO-3,4,PSO-1)
2. Explain the creation of new process, process accounting and process termination. (PO-1,2,4,PSO-1)
3. Illustrate the basic IPC issues and techniques in UNIX system programming (PO-1,2,4,PSO-3)
4. Explore Node.js and Angular features and create component based web pages using them (PO3,4,PSO-3)
5. Design Front-end web pages and connect to the Back-end Databases. (PO3,4,PSO-3)
Mini Project:

Software Development for Portable Device/Rich Internet Applications/Embedded Systems
Course Code: CS64 Credits: 0:0:4:2
Prerequisites: HTML, JAVA, Microprocessors Contact Hours: --
Course Coordinator/s: Dr. Anita K, Veena G S, Pramod Sunagar

Course Contents:

Software Development for Portable Devices:

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

Embedded Systems:

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

Reference Books:

Scheme for Self-Study Evaluation:

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

Course Outcomes (COs):

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

1. Design applications to work with basic SQLite or Fire Base database. (PO-1,2,3 PSO-2)
2. Design applications to work with Location based services and hardwires. (PO-1,2,3 PSO-2)
3. Analyze the complexity of building project involving multiple hardware software technologies. (PO-1,2,3,9,10 PSO-2)
4. Build independent modules of different hardware and software tools that can communicate and work together. (PO-2,3,9,10,11 PSO-2)
5. Build and evaluate integrated modules and study for their reliability and maintainability. (PO-2,3,9,10,11 PSO-2,3)
Mobile Computing

Course Code: CSE07                      Credits: 3:0:1:0
Contact Hours: 42+28

Prerequisites: Computer Networks
Course Coordinator/s: Dr. Monica R Mundada

Course Contents:

Unit I

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

Unit II
Wireless Networks – 2: GPRS
GPRS and Packet Data Network, GPRS Network Architecture, GPRS Network Operations, data Services in GPRS, Applications for GPRS, Billing and Charging in GPRS.

Wireless Networks – 3: CDMA, 3G and WiMAX
Spread Spectrum technology, IS-95, CDMA versus GSM, Wireless Data, Third Generation Networks, Applications on 3G, Introduction to WiMAX.

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

Unit IV
Mobile OS and Computing Environment

Unit V
Building, Mobile Internet Applications
Thin client: Architecture, the client, Middleware, messaging Servers, Processing a Wireless request, Wireless Applications Protocol (WAP) Overview, Wireless Languages: Markup Languages, HDML, WML, HTML, cHTML, XHTML, Voice XML.

J2ME
Introduction, CDC, CLDC, MIDP; Programming for CLDC, MIDlet model, Provisioning, MIDlet life-cycle, Creating new application, MIDlet event handling, GUI in MIDP, Low level GUI Components, Multimedia APIs; Communication in MIDP, Security.
**Text Books:**


**Reference Books:**


**Course Outcomes (COs):**

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

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

**List of Lab Exercises:**

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

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

Course Contents:

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

Unit II

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

Unit IV

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

Text Book:


Reference Books:

Course Outcomes (COs):

At the end of the course, students should be able to:
1. Explain the image formation process, pipeline architecture of computer graphics and the software and hardware components of a computer graphics system and basics of OpenGL API’s(PO-1,5,PSO-2)
2. Derive the geometrical transformations used in interactive computer graphics in different coordinate systems and for viewing and projections.(PO-1,5,12,PSO-2)
3. Discuss the different algorithms for clipping and rasterization of lines and polygons, and for hidden surface removal. (PO-2,PSO-2)
4. Explain different lighting and shading models. (PO-2,3,PSO-2)
5. Write 3D computer graphics applications in OpenGL using knowledge of display systems, image synthesis, and interactive control.(PO-1,3,5,12,PSO-2)

List of Lab Exercises:

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

Course Code: CSE09  
Credits: 3:0:0:1  
Contact Hours: 42

Prerequisites: Data communications and Computer networks  
Course Coordinator/s: Sanjeetha R

Course Contents:

**Unit I**

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

**Unit II**

**How SDN Works Contd.** - SDN controller core modules, SDN controller interfaces, Existing controller implementations, potential issues with the SDN Controller, SDN Applications, Alternate SDN Methods – SDN via APIs, Benefits and Limitations of SDN via APIs, SDN via hypervisor based overlay networks.  
**Self-Study:** Benefits and Limitations of SDN via APIs, OpenFlow Limitations.

**Unit III**

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


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

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

**Unit V**

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

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

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

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

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

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

**Text Book:**


**Reference Book:**


**Course Outcomes (COs):**

At the end of the course, students should be able to:
1. Describe the fundamental characteristics of SDN. (PO-1,2, PSO-3 )
2. Differentiate between various OpenFlow specifications. (PO-1,2,3,4, PSO-3 )
3. Identify different ways of implementing SDN. (PO-1,2,3, 4, PSO-3 )
4. Compare and contrast different types of controller models in SDN. (PO-1,2,3,4, PSO-3 )
5. Illustrate use of SDN in Data Centers and other environments. (PO-1,2,3, 4, PSO-3 )
Soft Computing

Course Code: CSE10
Prerequisites: Nil
Course Coordinator/s: Nagabhushan A M

Course Contents:

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

Unit II

Unit III

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

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

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

Reference Book:

Course Outcomes (COs):
At the end of the course, the student should be able to:
1. Identify and describe various problems on artificial neural networks and familiarize all concepts on various networks and applications in them. (PO-1,2,11, PSO-2)
2. Identify the compositions of neural networks, perceptrons and other networks. (PO-5,11, PSO-2)
3. Examine various fuzzification techniques and practice them. (PO-1,2, 4,11, PSO-2)
4. Design problems and obtain crisp values from fuzzy data using defuzzification. (PO-2,5,11, PSO-2)
5. Describe various compositions and complexities of genetic algorithms. (PO-1,4,11, PSO-2)
Machine Learning

Course Code: CSE11  
Prerequisites: AI  
Course Coordinator/s: Dr. S Rajarajeswari

Course Contents:

Unit I
Introduction

Unit II
Linear Models

Unit III
Tree and Probabilistic Models

Unit IV
Dimensionality Reduction and Evolutionary Models

Unit V
Graphical Models

Text Books:
Reference Books:

3. Trevor Hastie, Robert Tibshirani, Jerome Friedman, h The Elements of Statistical Learning, 2nd edition, springer series in statistics.

Course Outcomes (COs):

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

1. Distinguish between, supervised, unsupervised and semi-supervised learning. (PO1,PO2,PO4)
2. Apply the appropriate machine learning strategy for any given problem. (PO2,PO3,PO5,PO6,PO8)
3. Suggest tree and probabilistic learning algorithms for any given problem. (PO2,PO3,PO5)
4. Modify existing machine learning algorithms to improve classification efficiency. (PO3,PO4,PO5,PO6,PO7)
5. Design systems that uses the appropriate graph models of machine learning. (PO2,PO3,PO5,PO6,PO7)

List of Lab Exercises:

1. Introduction to python Numpy, scipy scikit and exercises.
2. Data sampling Visualization, Learning and Classification-Maximum margin classification, Classification errors, regularization, logistic regression.
3. Simple Linear regression, estimator bias and variance, active learning.
4. Dynamic Version spaces in ML.
5. Bayesian Decision Exercises.
7. Support vector machine (SVM) and kernels, kernel optimization.
10. Time series analysis.
11. Missing value updation, Error correction.
12. Elements of Linear discriminant functions.
13. Anomaly detection and recommendation.
Natural Language Processing

Course Code: CSE12  
Prerequisites: AI  
Course Coordinator/s: Dr. Jagadish S Kallimani

Credits: 3:0:1:0  
Contact Hours: 42+28

Course Contents:

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

Unit II

Unit III

Unit IV

Unit V

Text Book:


Reference Book:

Course Outcomes (COs):

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

1. Interpret how speech and language technology relies on formal models to capture knowledge, and language processing deals with subparts of words (morphology). (PO-1,5,11, PSO-2)
2. Illustrate the way N-gram tool is used for spelling and pronunciation processing, and part-of-speech tagging mechanism using various categories. (PO-2,3, 11, PSO-2)
3. Describe feature structures and unification operation which is used to combine them, and probabilistic parsing to capture more syntactic information. (PO-2,11, PSO-2)
4. Outline representations used to bridge the gap from language to commonsense Knowledge (semantic processing), and meanings associated with lexical items. (PO-1,3,5,11, PSO-2)
5. Emphasize problems that NLP systems face, natural language outputs construction from non-linguistic inputs and machine translation framework approaches. (PO-1,11, PSO-2)

List of Lab Exercises:

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

1. Writing Regular Expressions.
3. Compute Probabilities For Individual Words Of A Sentence.
4. Finite-State Automaton For A Dialogue Manager.
5. Identification And Disambiguation Of Polysemy Words And Others.
Information Retrieval

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

Course Contents:

Unit I

Unit II

Unit III

Unit IV

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

Text Books/ Reference Books:

Online Books:


Course Outcomes (COs):

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

1. Describe text operations and various information retrieval models. (PO-1,2, PSO-2)
2. Evaluate an IR system using various evaluation measures. (PO-1,2,4, PSO-2)
3. Apply various algorithms such as string matching, map reduce, classification and clustering. (PO-1,2,3,4,5, PSO-3)
4. Design web search engine, web crawling and link analysis techniques. (PO-1,2,3,4, PSO-3)
5. Explain XML Retrieval and various semantic web technologies.(PO-1, 3, 5, PSO-3)

List of Lab Exercises:

1. Implementation of string matching algorithms such as KMP and Rabin Karp.
2. Implementation of Construction of Inverted index.
3. Implementation of tf-idf calculations for sample text documents.
4. Implementation of Naïve Bayes Algorithm.
8. Exploring various ontology tools.
UNIX System Programming and Compiler Design Laboratory

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

Course Contents:

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

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

**Text Books:**

**Reference Books:**

**Course Outcomes (COs):**

At the end of the course, the student will be able to:
1. Build analyzers to recognize tokens and errors.(PO-1,2,4,PSO-2,3)
2. Construct generators for intermediate codes, optimizations and assembly language codes(PO-1,2,4,PSO-2,3)
3. Demonstrate advanced UNIX operating system concepts and terminology. (PO-1,2,4,PSO-2,3)
IOT/Embedded Systems Laboratory

Course Code: CSL67
Credit: 0:0:1:0
Contact Hours: 28

Prerequisites: Microprocessors/Microcontrollers
Course Coordinator/s: Dr. Diwakar Harekal

Course Contents:

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

Text Books:


Course Outcomes (COs):

At the end of the course, the student will be able to:
1. Understanding of use of GPIO interface with various devices directly. (PO-2,3,9,10,PSO-2)
2. Understanding of use of interface protocols like I2C, SPI and UART. (PO-2,3,9,10,11,PSO-2)
3. Work in a multidisciplinary project with varied interfaces and IDEs. (PO-2, 3, 9, 10, 11, PSO-2, 3).
Course Code: CSL68
Prerequisites: Nil
Course Coordinator/s: J Geetha

Course Contents:

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

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

Reference Books:

3. Web link for Angular4.0: https://angular.io/
5. Web link for MongoDB: https://www.mongodb.com

Course Outcomes (COs):

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

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