VISION OF THE DEPARTMENT:

To educate students with conceptual knowledge and technical skills in the field of Information Technology with moral and ethical values to achieve excellence in academic, industry and research centric environments.

MISSION OF THE DEPARTMENT:

1. To inculcate in students a firm foundation in theory and practice of IT skills coupled with the thought process for disruptive innovation and research methodologies, to keep pace with emerging technologies.
2. To provide a conducive environment for all academic, administrative and interdisciplinary research activities using state-of-the-art technologies.
3. To stimulate the growth of graduates and doctorates, who will enter the workforce as productive IT engineers, researchers and entrepreneurs with necessary soft skills, and continue higher professional education with competence in the global market.
4. To enable seamless collaboration with the IT industry and Government for consultancy and sponsored research.
5. To cater to cross-cultural, multinational and demographic diversity of students.
6. To educate the students on the social, ethical, and moral values needed to make significant contributions to society.
1. PROGRAMME EDUCATIONAL OBJECTIVES (PEOs):

1. To prepare students to excel in computer applications to succeed in industry/technical profession.
2. To provide students with solid foundation in mathematical and computing fundamentals and techniques required to solve technology related problems and also to pursue higher studies and research.
3. To train students with breadth of knowledge so as to comprehend, analyze, design and create computing solutions for the real life problems.
4. To inculcate a professional and ethical attitude in students, in order to work towards a broader social context.
5. To develop students with leadership qualities, and continuous learning ability on technology and trends needed for a successful career.

2. PROGRAMME OUTCOMES (POs):

After going through the three years of study, our Masters in Computer Applications Graduates will exhibit:

<table>
<thead>
<tr>
<th>PO#</th>
<th>Programme Outcome</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>An ability to independently carry out research/investigation and development work to solve practical problems.</td>
</tr>
<tr>
<td>2.</td>
<td>An ability to write and present a substantial technical report/document.</td>
</tr>
<tr>
<td>3.</td>
<td>An ability to demonstrate a degree of mastery over design and development of computer applications.</td>
</tr>
<tr>
<td>4.</td>
<td>An ability to create, select, adapt and apply appropriate innovative techniques, resources, and modern computing tools to complex computing activities with an understanding of the limitations.</td>
</tr>
<tr>
<td>5.</td>
<td>An ability to recognize the need and to engage in independent learning for continual development as a computing professional.</td>
</tr>
<tr>
<td>6.</td>
<td>An ability to function effectively as an individual and as a member/leader of a team in various technical environments.</td>
</tr>
</tbody>
</table>
3. PEO/PO Mapping:

<table>
<thead>
<tr>
<th>PROGRAMME EDUCATIONAL OBJECTIVES</th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>II</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>III</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>IV</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>V</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

Mapping of Course Outcome and Programme Outcome

<table>
<thead>
<tr>
<th>Year</th>
<th>Semester</th>
<th>Course Name</th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>YEAR 1</td>
<td>SEM 1</td>
<td>Mathematical foundations of Computer Science</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Digital Logic and Computer Organization</td>
<td></td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Problem Solving using Python</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Database Management Systems</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Research Methodology and IPR</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Programming in Python Laboratory</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Database Management Systems Laboratory</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>SEM 2</td>
<td>C Programming and Data Structures</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Operating Systems</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Software Engineering</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>YEAR 2</td>
<td>SEM 3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---------</td>
<td>--------------------------</td>
<td>----------------------</td>
<td>----------------------</td>
<td>----------------------</td>
<td>----------------------</td>
<td>----------------------</td>
<td>----------------------</td>
<td>----------------------</td>
</tr>
<tr>
<td></td>
<td>Advances in Databases</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>Web Programming</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>Personality Development</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Through Life Enlightenment Skills</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>C Programming and Data</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>Structures Laboratory</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Web Programming Laboratory</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>Computer Communication</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>and Networks</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Java Programming</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>Advanced Data Structures</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>and Algorithm Design</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Program Elective I</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Program Elective II</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Java Programming and</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>Networks Laboratory</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Advance Data Structures</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>and Algorithms Laboratory</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Socially Relevant Project</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>Data Science</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>Embedded Systems and</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Internet of Things</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>Distributed and Cloud</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>Computing</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Program Elective III</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Program Elective IV</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Program Elective V</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>YEAR 3</td>
<td>SEM 5</td>
<td>SEM 6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>-------</td>
<td>-------</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Project Work</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data Science Laboratory</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IoT and Cloud Laboratory</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>System Development Laboratory</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mobile Application Development Techniques</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Information Security</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cost Management of Engineering Projects(OEC)</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Program Elective VI (one from list of electives VI)</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Program Elective VII (one from list of electives VII)</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mobile Application development Laboratory</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Software Development Laboratory</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Summer Internship</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Project Work</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Semester I

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Code No.</th>
<th>Course Title</th>
<th>Category</th>
<th>Periods Per Week</th>
<th>Total Contact Periods</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Theory</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.</td>
<td>MA5102</td>
<td>Mathematical Foundations of Computer Science</td>
<td>FC</td>
<td>3 1 0</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>2.</td>
<td>CA5101</td>
<td>Digital Logic and Computer Organization</td>
<td>FC</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>3.</td>
<td>CA5102</td>
<td>Problem Solving using Python</td>
<td>PCC</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>4.</td>
<td>CA5103</td>
<td>Database Management Systems</td>
<td>PCC</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>5.</td>
<td>RM5151</td>
<td>Research Methodology and IPR</td>
<td>RMC</td>
<td>2 0 0</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>6.</td>
<td></td>
<td>Audit Course I*</td>
<td>AC</td>
<td>2 0 0</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Practicals</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7.</td>
<td>CA5111</td>
<td>Programming in Python Laboratory</td>
<td>PCC</td>
<td>0 0 4</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>8.</td>
<td>CA5112</td>
<td>Database Management Systems Laboratory</td>
<td>PCC</td>
<td>0 0 4</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Total</strong></td>
<td></td>
<td>16 1 8</td>
<td>25</td>
<td>19</td>
</tr>
</tbody>
</table>

*Audit course is optional

### Semester II

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Code No.</th>
<th>Course Title</th>
<th>Category</th>
<th>Periods Per Week</th>
<th>Total Contact Periods</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Theory</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.</td>
<td>CA5201</td>
<td>C Programming and Data Structures</td>
<td>PCC</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>2.</td>
<td>CA5202</td>
<td>Operating Systems</td>
<td>PCC</td>
<td>3 0 2</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>3.</td>
<td>CA5203</td>
<td>Software Engineering</td>
<td>PCC</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>4.</td>
<td>CA5204</td>
<td>Advances in Databases</td>
<td>PCC</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>5.</td>
<td>CA5205</td>
<td>Web Programming</td>
<td>PCC</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>6.</td>
<td></td>
<td>Audit Course II*</td>
<td>AC</td>
<td>2 0 0</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Practicals</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7.</td>
<td>CA5211</td>
<td>C Programming and Data Structures Laboratory</td>
<td>PCC</td>
<td>0 0 4</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>8.</td>
<td>CA5212</td>
<td>Web Programming Laboratory</td>
<td>PCC</td>
<td>0 0 4</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Total</strong></td>
<td></td>
<td>17 0 10</td>
<td>27</td>
<td>20</td>
</tr>
</tbody>
</table>

*Audit course is optional
### SEMESTER III

<table>
<thead>
<tr>
<th>S. NO.</th>
<th>CODE NO.</th>
<th>COURSETITLE</th>
<th>CATEGORY</th>
<th>PERIODS PER WEEK</th>
<th>TOTAL CONTACT PERIODS</th>
<th>CREDITS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>L</td>
<td>T</td>
<td>P</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>THEORY</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.</td>
<td>CA5301</td>
<td>Computer Communication and Networks</td>
<td>PCC</td>
<td>3</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>2.</td>
<td>CA5302</td>
<td>Java Programming</td>
<td>PCC</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3.</td>
<td>CA5303</td>
<td>Advanced Data Structures and Algorithm Design</td>
<td>PCC</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4.</td>
<td></td>
<td>Program Elective I</td>
<td>PEC</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5.</td>
<td></td>
<td>Program Elective II</td>
<td>PEC</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td><strong>PRACTICALS</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6.</td>
<td>CA5311</td>
<td>Java Programming and Networks Laboratory</td>
<td>PCC</td>
<td>0</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>7.</td>
<td>CA5312</td>
<td>Advanced Data Structures and Algorithms Laboratory</td>
<td>PCC</td>
<td>0</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>8.</td>
<td>CA5313</td>
<td>Socially Relevant Project</td>
<td>EEC</td>
<td>0</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td><strong>TOTAL</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>15</td>
<td>1</td>
<td>10</td>
</tr>
</tbody>
</table>

### SEMESTER IV

<table>
<thead>
<tr>
<th>S. NO.</th>
<th>CODE NO.</th>
<th>COURSETITLE</th>
<th>CATEGORY</th>
<th>PERIODS PER WEEK</th>
<th>TOTAL CONTACT PERIODS</th>
<th>CREDITS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>L</td>
<td>T</td>
<td>P</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>THEORY</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.</td>
<td>CA5401</td>
<td>Data Science</td>
<td>PCC</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2.</td>
<td>CA5402</td>
<td>Embedded Systems and Internet of Things</td>
<td>PCC</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3.</td>
<td>CA5403</td>
<td>Cloud Computing</td>
<td>PCC</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4.</td>
<td></td>
<td>Program Elective III</td>
<td>PEC</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5.</td>
<td></td>
<td>Program Elective IV</td>
<td>PEC</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>6.</td>
<td></td>
<td>Program Elective V</td>
<td>PEC</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td><strong>PRACTICALS</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7.</td>
<td>CA 5411</td>
<td>Data Science Laboratory</td>
<td>PCC</td>
<td>0</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>8.</td>
<td>CA 5412</td>
<td>Internet of Things and Cloud Laboratory</td>
<td>PCC</td>
<td>0</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>9.</td>
<td>CA 5413</td>
<td>System Development Laboratory</td>
<td>EEC</td>
<td>0</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td><strong>TOTAL</strong></td>
<td></td>
<td></td>
<td></td>
<td>18</td>
<td>0</td>
<td>10</td>
</tr>
</tbody>
</table>
## SEMESTER V

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Code No.</th>
<th>Course Title</th>
<th>Category</th>
<th>Periods Per Week</th>
<th>Total Contact Periods</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>THEORY</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.</td>
<td>CA5501</td>
<td>Mobile Application Development Techniques</td>
<td>PCC</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>2.</td>
<td>CA5502</td>
<td>Information Security</td>
<td>PCC</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>3.</td>
<td></td>
<td>Open Elective</td>
<td>OEC</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>4.</td>
<td></td>
<td>Program Elective VI</td>
<td>PEC</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>5.</td>
<td></td>
<td>Program Elective VII</td>
<td>PEC</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PRACTICALS</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6.</td>
<td>CA5511</td>
<td>Mobile Application Development Laboratory</td>
<td>PCC</td>
<td>0 0 4</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>7.</td>
<td>CA5512</td>
<td>Software Development Laboratory</td>
<td>PCC</td>
<td>0 0 4</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>8.</td>
<td>CA5513</td>
<td>Summer Internship</td>
<td>EEC</td>
<td>0 0 2</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>TOTAL</strong></td>
<td></td>
<td>15 0 10</td>
<td>25</td>
<td>20</td>
</tr>
</tbody>
</table>

## SEMESTER VI

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Code No.</th>
<th>Course Title</th>
<th>Category</th>
<th>Periods Per Week</th>
<th>Total Contact Periods</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>PRACTICALS</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.</td>
<td>CA5611</td>
<td>Project Work</td>
<td>EEC</td>
<td>0 0 24</td>
<td>24</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>TOTAL</strong></td>
<td></td>
<td>0 0 24</td>
<td>24</td>
<td>12</td>
</tr>
</tbody>
</table>

## FOUNDATION COURSES (FC)

<table>
<thead>
<tr>
<th>SL. No</th>
<th>Code No.</th>
<th>Course Title</th>
<th>Periods Per Week</th>
<th>Credits</th>
<th>Semester</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>MA5102</td>
<td>Mathematical Foundations of Computer Science</td>
<td>3 1 0</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>2.</td>
<td>CA5101</td>
<td>Digital Logic and Computer Organization</td>
<td>3 0 0</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>TOTAL CREDITS</strong></td>
<td></td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>SL.NO</td>
<td>CODE_NO.</td>
<td>COURSE TITLE</td>
<td>PERIODS PER WEEK</td>
<td>CREDITS</td>
<td>SEMESTER</td>
</tr>
<tr>
<td>-------</td>
<td>----------</td>
<td>--------------------------------------------------</td>
<td>------------------</td>
<td>---------</td>
<td>----------</td>
</tr>
<tr>
<td>1.</td>
<td>CA5102</td>
<td>Problem Solving using Python</td>
<td>3 0 0</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>2.</td>
<td>CA5103</td>
<td>Database Management Systems</td>
<td>3 0 0</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>3.</td>
<td>CA5111</td>
<td>Programming in Python Laboratory</td>
<td>0 0 4</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>4.</td>
<td>CA5112</td>
<td>Database Management Systems Laboratory</td>
<td>0 0 4</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>5.</td>
<td>CA5201</td>
<td>C Programming and Data Structures</td>
<td>3 0 0</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>6.</td>
<td>CA5202</td>
<td>Operating Systems</td>
<td>3 0 2</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>7.</td>
<td>CA5203</td>
<td>Software Engineering</td>
<td>3 0 0</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>8.</td>
<td>CA5204</td>
<td>Advances in Databases</td>
<td>3 0 2</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>9.</td>
<td>CA5205</td>
<td>Web Programming</td>
<td>3 0 0</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>10.</td>
<td>CA5211</td>
<td>C Programming and Data Structures Laboratory</td>
<td>0 0 4</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>11.</td>
<td>CA5212</td>
<td>Web Programming Laboratory</td>
<td>0 0 4</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>12.</td>
<td>CA5301</td>
<td>Computer Communication and Networks</td>
<td>3 1 0</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>13.</td>
<td>CA5302</td>
<td>Java Programming</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>14.</td>
<td>CA5303</td>
<td>Advanced Data Structures and Algorithm Design</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>15.</td>
<td>CA5311</td>
<td>Java Programming and Networks Laboratory</td>
<td>0 0 4</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>16.</td>
<td>CA5312</td>
<td>Advanced Data Structures and Algorithms Laboratory</td>
<td>0 0 4</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>17.</td>
<td>CA5401</td>
<td>Data Science</td>
<td>3 0 0</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>18.</td>
<td>CA5402</td>
<td>Embedded Systems and Internet of Things</td>
<td>3 0 0</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>19.</td>
<td>CA5403</td>
<td>Cloud Computing</td>
<td>3 0 0</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>20.</td>
<td>CA5411</td>
<td>Data Science Laboratory</td>
<td>0 0 4</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>21.</td>
<td>CA5412</td>
<td>Internet of Things and Cloud Laboratory</td>
<td>0 0 4</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>22.</td>
<td>CA5413</td>
<td>System Development Laboratory</td>
<td>3 0 0</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>23.</td>
<td>CA5502</td>
<td>Information Security</td>
<td>3 0 0</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>24.</td>
<td>CA5501</td>
<td>Mobile Application Development</td>
<td>3 0 0</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>S.NO</td>
<td>CODE NO.</td>
<td>COURSE TITLE</td>
<td>PERIODS PER WEEK</td>
<td>CREDITS</td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>----------</td>
<td>--------------------------------------------------</td>
<td>------------------</td>
<td>---------</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>CA5001</td>
<td>Blockchain Technologies</td>
<td>PEC 3 0 0 0</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>CA5002</td>
<td>Ethical Hacking</td>
<td>PEC 3 0 0 0</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>CA5003</td>
<td>Big Data with R</td>
<td>PEC 3 0 0 0</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>CA5004</td>
<td>Full Stack Development</td>
<td>PEC 3 0 0 0</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>CA5005</td>
<td>Introduction to Machine Learning</td>
<td>PEC 3 0 0 0</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>CA5006</td>
<td>Autonomous Ground Vehicle Systems</td>
<td>PEC 3 0 0 0</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>CA5007</td>
<td>E-Learning Techniques</td>
<td>PEC 3 0 0 0</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>CA5008</td>
<td>Software Testing</td>
<td>PEC 3 0 0 0</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>CA5009</td>
<td>Deep Learning Techniques and Applications</td>
<td>PEC 3 0 0 0</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>CA5010</td>
<td>Game Programming Techniques</td>
<td>PEC 3 0 0 0</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>CA5011</td>
<td>Multimedia Technologies</td>
<td>PEC 3 0 0 0</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>CA5012</td>
<td>Data Visualization Techniques</td>
<td>PEC 3 0 0 0</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>CA5013</td>
<td>UNIX Internals</td>
<td>PEC 3 0 0 0</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>CA5014</td>
<td>C# and .NET Programming</td>
<td>PEC 3 0 0 0</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>CA5015</td>
<td>Service Oriented Architectures</td>
<td>PEC 3 0 0 0</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>CA5016</td>
<td>Software Project Management</td>
<td>PEC 3 0 0 0</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>CA5017</td>
<td>Mixed Reality</td>
<td>PEC 3 0 0 0</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>CA5018</td>
<td>Digital Image Processing and Applications</td>
<td>PEC 3 0 0 0</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>CA5019</td>
<td>Text Mining Techniques</td>
<td>PEC 3 0 0 0</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>CA5020</td>
<td>Data Warehousing and Data Mining Techniques</td>
<td>PEC 3 0 0 0</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>CA5021</td>
<td>Software Quality Assurance</td>
<td>PEC 3 0 0 0</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>CA5022</td>
<td>Introduction to Social Network Analysis</td>
<td>PEC 3 0 0 0</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>CA5023</td>
<td>IoT Based Smart Systems</td>
<td>PEC 3 0 0 0</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>CA5024</td>
<td>Object Oriented Analysis and Design</td>
<td>PEC 3 0 0 0</td>
<td>3</td>
<td></td>
</tr>
</tbody>
</table>
### RESEARCH METHODOLOGY AND IPR COURSES (RMC)

<table>
<thead>
<tr>
<th>Sl. No</th>
<th>Course Code</th>
<th>Course Title</th>
<th>Category</th>
<th>Periods per Week</th>
<th>Credits</th>
<th>Semester</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CA5025</td>
<td>Artificial Intelligence</td>
<td></td>
<td>3 0 0</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>CA5026</td>
<td>Computer Graphics</td>
<td></td>
<td>3 0 0</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>26</td>
<td>CA5027</td>
<td>Human Computer Interaction</td>
<td></td>
<td>3 0 0</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>28</td>
<td>CA5028</td>
<td>Wireless Sensor Networks &amp; Protocols</td>
<td></td>
<td>3 0 0</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>29</td>
<td>CA5029</td>
<td>Next Generation Networks</td>
<td></td>
<td>3 0 0</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>30</td>
<td>CA5030</td>
<td>Cybernetics</td>
<td></td>
<td>3 0 0</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>31</td>
<td>CA5031</td>
<td>Network Programming and Management</td>
<td></td>
<td>3 0 0</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>32</td>
<td>CA5032</td>
<td>Semantic Web and Applications</td>
<td></td>
<td>3 0 0</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>33</td>
<td>CA5033</td>
<td>Soft Computing</td>
<td></td>
<td>3 0 0</td>
<td>3</td>
<td>1</td>
</tr>
</tbody>
</table>

**Total Credits:** 2

### OPEN ELECTIVE COURSES (OEC)

*(out of 6 courses one course must be selected)*

<table>
<thead>
<tr>
<th>Sl. No</th>
<th>Course Code</th>
<th>Course Title</th>
<th>Category</th>
<th>Periods per Week</th>
<th>Contact Periods</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>OE5091</td>
<td>Business Data Analytics</td>
<td>OEC</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>OE5092</td>
<td>Industrial Safety</td>
<td>OEC</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>OE5093</td>
<td>Operations Research</td>
<td>OEC</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>OE5094</td>
<td>Cost Management of Engineering Projects</td>
<td>OEC</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>OE5095</td>
<td>Composite Materials</td>
<td>OEC</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>6</td>
<td>OE5096</td>
<td>Waste to Energy</td>
<td>OEC</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
</tbody>
</table>
AUDIT COURSES (AC)
Registration for any of these courses is optional to students.

<table>
<thead>
<tr>
<th>Sl. No</th>
<th>Course Code</th>
<th>Course Title</th>
<th>Periods per Week</th>
<th>Credits</th>
<th>Semester</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>AX5091</td>
<td>English for Research Paper Writing</td>
<td>Lecture: 2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2.</td>
<td>AX5092</td>
<td>Disaster Management</td>
<td>Lecture: 2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3.</td>
<td>AX5093</td>
<td>Sanskrit for Technical Knowledge</td>
<td>Lecture: 2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4.</td>
<td>AX5094</td>
<td>Value Education</td>
<td>Lecture: 2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5.</td>
<td>AX5095</td>
<td>Constitution of India</td>
<td>Lecture: 2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>6.</td>
<td>AX5096</td>
<td>Pedagogy Studies</td>
<td>Lecture: 2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>7.</td>
<td>AX5097</td>
<td>Stress Management by Yoga</td>
<td>Lecture: 2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>8.</td>
<td>AX5098</td>
<td>Personality Development Through Life Enlightenment Skills</td>
<td>Lecture: 2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>9.</td>
<td>AX5099</td>
<td>Unnat Bharat Abhiyan</td>
<td>Lecture: 2</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Total Credits: 0

EMPLOYABILITY ENHANCEMENT COURSES (EEC)

<table>
<thead>
<tr>
<th>Sl. No</th>
<th>Code No.</th>
<th>Course Title</th>
<th>Periods per Week</th>
<th>Credits</th>
<th>Semester</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>CA5313</td>
<td>Socially Relevant Project</td>
<td>Lecture: 0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>2.</td>
<td>CA 5413</td>
<td>System Development Laboratory</td>
<td>Lecture: 0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>3.</td>
<td>CA5513</td>
<td>Summer Internship</td>
<td>Lecture: 0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>4.</td>
<td>CA5611</td>
<td>Project Work</td>
<td>Lecture: 0</td>
<td>0</td>
<td>12</td>
</tr>
</tbody>
</table>

Total Credits: 15

<table>
<thead>
<tr>
<th>S. No</th>
<th>Subject Area</th>
<th>Name of the Program</th>
<th>Credits per Semester</th>
<th>Credits Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>FC</td>
<td></td>
<td>I</td>
<td>II</td>
</tr>
<tr>
<td>2</td>
<td>PCC</td>
<td></td>
<td>10</td>
<td>21</td>
</tr>
<tr>
<td>3</td>
<td>PEC</td>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>RMC</td>
<td></td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>OEC</td>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>EEC</td>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>Non Credit / Audit Course</td>
<td></td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Total Credit: 19 | 21 | 21 | 23 | 20 | 12 | 116
OBJECTIVES:
- To introduce Mathematical Logic, Inference Theory and proof methods.
- To provide fundamental principles of combinatorial counting techniques.
- To explain the fundamental algebraic structures, groups and their algebraic properties.
- To provide exposure to the development of the algebraic structures, lattices and Boolean algebra and to demonstrate the utility of Boolean laws.
- To familiarize Finite State Automata and Context-free grammar which are used for formal language generation.

UNIT I LOGIC 12

UNIT II COMBINATORICS 12
Permutations and Combinations - Mathematical Induction - Pigeonhole principle - Principle of Inclusion and Exclusion - Recurrence relations - Solution by generating functions and characteristics equations.

UNIT III ALGEBRAIC STRUCTURES 12
Groups - Cyclic group - Permutation group (S_n and D_n) - Substructures - Homomorphism - Cosets and Lagrange’s Theorem - Normal Subgroups - Rings and Fields (definition and examples).

UNIT IV LATTICES 12
Partial order relation – Posets - Hasse diagram - Lattices - Special Lattices - Boolean Algebra.

UNIT V FINITE STATE AUTOMATA AND GRAMMARS 12
Finite state automata - Deterministic and non-deterministic model - languages accepted by Finite State Automata - Regular expressions - Context-free grammars - Derivation trees.

TOTAL: 60 PERIODS

OUTCOMES:
At the end of the course, students will be able to
- Understand the validity of the logical arguments, mathematical proofs and correctness of the algorithm.
- Apply combinatorial counting techniques in solving combinatorial related problems.
- Understand the significance of algebraic structural ideas used in coding theory and cryptography.
- Apply Boolean laws and Boolean functions in combinatorial circuit designs.
- Construct Finite State Automation for constructing regular sets as well as context-free grammar to generate context-free language.

REFERENCES:
OBJECTIVES:
- To understand the fundamentals of Boolean logic and functions.
- To design and realize digital systems with basic gates and other components using combinational and sequential circuits.
- To study the instruction sets and operations of a processor.
- To study the different ways of communication with I/O devices and standard I/O Interfaces.
- To study the hierarchical memory system including cache memories and virtual memory.

UNIT I  DIGITAL FUNDAMENTALS  9

Suggested Activities:
- Flipped classroom on value systems.
- Proofs and simplification in class.
- Practical - Implementation of simple functions using gates.

Suggested Evaluation Methods:
- Quizzes on number systems and conversions.
- Mock test on Boolean simplifications.

UNIT II  COMBINATIONAL AND SEQUENTIAL CIRCUITS  9

Suggested Activities:
- Flipped classroom on analysis of combinational circuits.
- External learning - Introduction to propositional problems using conjunction, disjunction and negation.
- Practical - Implementation of simple functions using gates.

Suggested Evaluation Methods:
- Assignment on simplifying and implementing Boolean function using Multiplexer and decoders.
- Mock test for solving problems on designing counters.
- Quizzes on encoder, decoder and other topics of the unit.

UNIT III  COMPUTER FUNDAMENTALS  9

Suggested Activities:
- Flipped classroom on evolution and types of computer systems, identification of benchmarks.
- Practical - Installing and using simulator for RISC and CISC.
- Mapping and correlating a C code with its machine code.
- Practical - Opening up a computer system and studying the components.

**Suggested Evaluation Methods:**
- Mock Test on processor performance problems.
- Practical - Analyzing the ISA supported by the architectural simulator and running simple programs on the simulator and quizzes for evaluation.
- Quizzes on classes of architecture.

**UNIT IV PROCESSOR**

Instruction Execution – Building a Data Path – Designing a Control Unit – Hardwired Control, Microprogrammed Control – Pipelining – Data Hazard – Control Hazards.

**Suggested Activities:**
- Flipped classroom on evolution of processor architecture.
- Tutorial for identifying and classifying hazards in code snippet.
- Case study of the ARM and Intel processors.

**Suggested Evaluation Methods:**
- Quizzes on designing control unit.
- Mock test on identifying hazards in code snippet.

**UNIT V MEMORY AND I/O**


**Suggested Activities:**
- Flipped classroom on types of memory.
- Practical - Implementing a simple functional model for memory mapping in cache using C/C++.
- Discussion on hit/miss rates for various access patterns. Experimenting with different replacement policies.
- Case study of the memory hierarchy of ARM Cortex and Intel i7.

**Suggested Evaluation Methods:**
- Assignment on memory management.
- Quizzes on I/O.

**TOTAL: 45 PERIODS**

**OUTCOMES:**
On completion of the course, the students will be able to:
- Be proficient in number systems and computer arithmetic.
- Design and implement digital systems with basic gates and other components using combinational and sequential circuits.
- Familiarize and understand the organization of memory hierarchies including the basics of cache design and subsystem.
- Understand a machine’s Instruction Set Architecture (ISA) including basic instruction fetch and execute cycles, instruction formats and control flow.
- Understand a basic input/output functioning including program controlled I/O and interrupt I/O.
- Analyze the performance of processors and caches.
REFERENCES:

<table>
<thead>
<tr>
<th></th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>CO2</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>CO3</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>CO4</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>CO5</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>CO6</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

CA5102 PROBLEM SOLVING USING PYTHON L T P C 3 0 0 3

OBJECTIVES:
- To know the basics of algorithmic problem solving.
- To develop Python programs with conditionals and loops.
- To define Python functions and use function calls.
- To use Python data structures – lists, tuples, dictionaries.
- To do input/output with files in Python.

UNIT I INTRODUCTION TO COMPUTING AND PROBLEM SOLVING 9

Suggested Activities:
- Developing pseudocodes and flowcharts for real life activities such as railway ticket booking using IRCTC, admission process to undergraduate course, academic schedules during a semester etc.
- Developing algorithms for basic mathematical expressions using arithmetic operations.
- Installing Python.
- Simple programs on print statements, arithmetic operations.
Suggested Evaluation Methods:
- Assignments on pseudocodes and flowcharts.
- Tutorials on Python programs.

UNIT II  CONDITIONALS AND FUNCTIONS

Suggested Activities:
- Simple Python program implementation using Operators, Conditionals, Iterative Constructs and Functions.
- Practical - Implementation of a simple calculator.
- Developing simple applications like calendar, phone directory, to-do lists etc.
- Flow charts for GCD, Exponent Functions, Fibonacci Series using conditionals and iterative statements.
- External learning - Recursion vs. Iteration.

Suggested Evaluation Methods:
- Tutorials for the above activities.
- Group discussion on external learning.

UNIT III  SIMPLE DATA STRUCTURES IN PYTHON

Suggested Activities:
- Practical - Implementing python program using lists, tuples, sets for the following scenario:
  - Simple Sorting techniques
  - Student Examination Report
  - Billing Scheme during Shopping
- External learning - List versus Tuple versus Set
- Practical - Implementing any application using the three data structures, list, tuple and set.

Suggested Evaluation Methods:
- Tutorials for the above activities.
- Group discussion on external learning component.

UNIT IV  STRINGS, DICTIONARIES, MODULES
Suggested Activities:

- Practical - Implementing Python program by importing Time module, Math package, etc.
- Creation of your own package and importing into the application.

Suggested Evaluation Methods:

- Tutorials for the above activities.

UNIT V
FILE HANDLING AND EXCEPTION HANDLING

Suggested Activities:

- Develop modules using Python to handle files and all operations on files.
- Usage of exceptions, multiple except blocks for applications that use delimiters like age, range of numerals, etc.
- Practical - Implementing Python program to open non-existent file using exceptions.

Suggested Evaluation Methods:

- Tutorials for the above activities.
- Case Studies.

TOTAL: 45 PERIODS

OUTCOMES:

On completion of the course, students will be able to:

1. Develop algorithmic solutions to simple computational problems.
2. Develop and execute simple Python programs.
4. Decompose a Python program into functions.
5. Represent compound data using Python lists, tuples, dictionaries.
6. Read and write data from/to files in Python Programs.

REFERENCES:

OBJECTIVES:

- To learn the fundamentals of data models, conceptualize and depict a database system using ER diagram.
- To study the principles to be followed to create an effective relational database and write SQL queries to store/retrieve data to/from database systems.
- To know the fundamental concepts of transaction processing, concurrency control techniques and recovery procedure.
- To learn about the internal storage structures using different file and indexing techniques and the basics of query processing and optimization.
- To study the basics of distributed databases, semi-structured and un-structured data models.

UNIT I       RELATIONAL DATABASES


Suggested Activities:

- Creating tables with key constraints, adding and removing constraints with referential integrity using DDL commands.
- Flipped classroom on relational algebra operations (selection, projection, joins etc.).
- Write SQL queries for demonstrating CRUD operations, aggregate functions and various join operations using DML commands.
- Create stored procedures for executing complex SQL transactions.
- Create triggers for alerting user/system while manipulating data.

Suggested Evaluation Methods:

- Tutorials on DDL, DML and DCL queries.
- Quizzes on relational algebra operations.
- Demonstration of created stored procedures and triggers.
UNIT II DATABASE DESIGN

Suggested Activities:
- Simple database application design using ER diagram.
- Practical - ER modeling using open source tools and realizing database.
- Study of various anomalies and normalizing table (1NF, 2NF, 3NF, BCNF).
- Flipped classroom on topics of database design and normalization.

Suggested Evaluation Methods:
- Tutorials on application specific ER Diagram.
- Tutorials on normalization and database design.

UNIT III TRANSACTION MANAGEMENT

Suggested Activities:
- Checking serializability among transactions.
- Flipped classroom on concurrency control protocols.
- Study of crash recovery algorithm (ARIES).

Suggested Evaluation Methods:
- Tutorials on serializability and crash recovery algorithm
- Quizzes on concurrency control protocols.

UNIT IV IMPLEMENTATION TECHNIQUES

Suggested Activities:
- Study of different RAID levels and its uses in different applications.
- Practical - Creation of B+ tree with insertion and deletion operations.
- Assignments on cost estimation of different types of queries.

Suggested Evaluation Methods:
- Report on applications of RAID levels.
- Tutorials on B+ Tree manipulation.
- Quizzes on hashing mechanisms.
- Exercise on cost estimation for various SQL queries.
- Evaluation of the practical assignments.
UNIT V  ADVANCED TOPICS

Suggested Activities:
- Design of distributed database using fragmentation.
- Creation of XML document based on XML schema.
- Creation of document and column oriented databases and simple manipulation.

Suggested Evaluation Methods:
- Tutorials on fragmenting database tables and writing simple SQL queries.
- Assignments on creation of XML schema and validating XML documents.
- Demonstration of created document and column-oriented databases.

TOTAL: 45 PERIODS

OUTCOMES:
On completion of the course, the student will be able to:
1. Model an application’s data requirements using conceptual modeling and design database schemas based on the conceptual model.
2. Formulate solutions to a broad range of query problems using relational algebra/SQL.
3. Demonstrate an understanding of normalization theory and apply such knowledge to the normalization of a database.
4. Run transactions and estimate the procedures for controlling the consequences of concurrent data access.
5. Explain basic database storage structures, access techniques and query processing.
6. Describe distributed, semi-structured and unstructured database systems.

REFERENCES:

<table>
<thead>
<tr>
<th></th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>CO2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>CO3</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>CO4</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>CO5</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>CO6</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
</tbody>
</table>
OBJECTIVES:
To impart knowledge and skills required for research and IPR:
- Problem formulation, analysis and solutions.
- Technical paper writing / presentation without violating professional ethics
- Patent drafting and filing patents.

UNIT I RESEARCH PROBLEM FORMULATION
Meaning of research problem- Sources of research problem, criteria characteristics of a good research problem, errors in selecting a research problem, scope and objectives of research problem. Approaches of investigation of solutions for research problem, data collection, analysis, interpretation, necessary instrumentations

UNIT II LITERATURE REVIEW
Effective literature studies approaches, analysis, plagiarism, and research ethics.

UNIT III TECHNICAL WRITING / PRESENTATION
Effective technical writing, how to write report, paper, developing a research proposal, format of research proposal, a presentation and assessment by a review committee.

UNIT IV INTRODUCTION TO INTELLECTUAL PROPERTY RIGHTS (IPR)

UNIT V INTELLECTUAL PROPERTY RIGHTS (IPR)

TOTAL: 30 PERIODS

OUTCOMES:
1. Ability to formulate research problem
2. Ability to carry out research analysis
3. Ability to follow research ethics
4. Ability to understand that today's world is controlled by Computer, Information Technology, but tomorrow world will be ruled by ideas, concept, and creativity
5. Ability to understand about IPR and filing patents in R & D.

<table>
<thead>
<tr>
<th></th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
<th>PO7</th>
<th>PO8</th>
<th>PO9</th>
<th>PO10</th>
<th>PO11</th>
<th>PO12</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CO2</td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CO3</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CO4</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CO5</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
</tbody>
</table>
REFERENCES:

CA5111 PROGRAMMING IN PYTHON LABORATORY

OBJECTIVES:
- To understand the problem solving approaches.
- To learn the basic programming constructs in Python.
- To articulate where computing strategies support in providing Python-based solutions to real world problems.
- To use Python data structures - lists, tuples, dictionaries.
- To do input/output with files in Python.

EXPERIMENTS:
1. Identification and solving of simple real life or scientific or technical problems, and developing flow charts for the same.
2. Python programming using simple statements and expressions.
3. Scientific problems using Conditionals and Iterative loops.
4. Practical - Implementing real-time/technical applications using Lists, Tuples.
5. Practical - Implementing real-time/technical applications using Sets, Dictionaries.
6. Practical - Implementing programs using Functions.
7. Practical - Implementing programs using Strings.
10. Practical - Implementing real-time/technical applications using Exception handling.
12. Developing a game activity using Pygame like bouncing ball, car race etc.

OUTCOMES:
On completion of the course, students will be able to:
1. Develop algorithmic solutions to simple computational problems
2. Develop and execute simple Python programs.
4. Decompose a Python program into functions.
5. Represent compound data using Python data structures.
6. Apply Python features in developing software applications.

<table>
<thead>
<tr>
<th></th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>CO2</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>CO3</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>CO4</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>CO5</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>CO6</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
</tr>
</tbody>
</table>
OBJECTIVES:
- To learn and implement important commands in SQL.
- To learn the usage of nested and joint queries.
- To understand functions, procedures and procedural extensions of databases.
- To understand design and implementation of typical database applications.
- To be familiar with the use of a front end tool for GUI based application development.

LABORATORY EXERCISES:
1. Create a database table, add constraints (primary key, unique, check, Not null), insert rows, update and delete rows using SQL DDL and DML commands.
2. Create set of tables, add foreign key constraints and incorporate referential integrity.
3. Query the database tables using different ‘where’ clause conditions and also implement aggregate functions.
4. Query the database tables and explore sub queries and simple join operations.
5. Query the database tables and explore natural, equi and outer joins.
6. Write user defined functions and stored procedures in SQL.
7. Execute complex transactions and realize DCL and TCL commands.
8. Write SQL Triggers for insert, delete, and update operations in database table.
9. Create View and index for database tables with large number of records.
10. Create a XML database and validate it using XML schema.
12. Develop a simple GUI based database application and incorporate all the above-mentioned features.

TOTAL: 60 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
1. Create databases with different types of key constraints.
2. Write simple and complex SQL queries using DML and DCL commands.
3. Realize database design using 3NF and BCNF.
4. Use advanced features such as stored procedures and triggers and incorporate in GUI based application development.
5. Create XML database and validate with meta-data (XML schema).
6. Create and manipulate data using NOSQL database.

<table>
<thead>
<tr>
<th></th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>CO2</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>CO3</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>CO4</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>CO5</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>CO6</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
</tr>
</tbody>
</table>
OBJECTIVES:
- To study the basics and advanced concepts of C programming language.
- To learn the concepts of linear data structures and its applications.
- To understand the concepts of non-linear data structures.
- To learn the usage of sorting techniques.
- To familiarize the concepts of hashing.

UNIT I  BASICS OF C PROGRAMMING
7
Data Types – Variables – Operators and Expressions – Conditional Statements – Control Statements – Arrays.

Suggested Activities:
- Flipped classroom on fundamentals of C Programming.
- External learning - Multidimensional arrays.
- Practical - Implementation of basic concepts of C programming like operators, conditional statements, loops etc.
- Practical - Implementation of programs using single-dimensional and multi-dimensional arrays.

Suggested Evaluation Methods:
- Quizzes on basic concepts of C Programming.
- Assignments on file handling and macros.
- Demonstration for practical learning implementations.

UNIT II  ADVANCED C PROGRAMMING
8

Suggested Activities:
- Flipped classroom on functions.
- External learning - Functions, pointers and macros.
- Practical - Implementation of programs using Call by Value and Call by Reference mechanism in functions.
- Practical - Implementation of programs using pointer to a functions, structure and memory allocation operators.
- Practical - Implementation of programs using recursion.

Suggested Evaluation Methods:
- Quizzes on C functions and pointers.
- Assignments on file handling and macros.
- Demonstration of the practical implementations.

UNIT III  LINEAR DATA STRUCTURES
9

Suggested Activities:
- Flipped classroom on basics of ADT’s.
- External learning - Cursor based implementation of linked lists, applications of lists, double ended queue.
• Practical - Implementation of Tower of Hanoi using Recursion.
• Practical - Implementation of Polynomial ADT using Lists.
• Practical - Implementation of the Evaluation of expression using Stack ADT.
• Practical - Implementation of any one application of Queue.

Suggested Evaluation Methods:
• Quizzes on ADTs.
• Assignments on double ended queues, applications of lists.
• Demonstration of the practical implementations.

UNIT IV HIERARCHICAL DATA STRUCTURES


Suggested Activities:
• Flipped classroom on fundamentals of non-linear data structures.
• External learning - Operations on binary search tree, complete binary tree.
• Practical - Implementation of operations such as counting the number of nodes in a BST, finding predecessor and successor of a given node, second largest node in a BST, finding the mirror image of a given tree etc.

Suggested Evaluation Methods:
• Quizzes on fundamentals of non-linear data structures.
• Assignments on complete binary tree.
• Demonstration for practical implementations.

UNIT V HASHING AND SORTING


Suggested Activities:
• Flipped classroom on hashing and hash functions, different sorting techniques such as Bubble Sort, Selection Sort etc.
• External learning - Search algorithms, priority queues, external sorting, replacement selection technique.
• Designing an efficient hash functions for a given problem.
• Practical - Solving a search problem in O(1) time using hashing technique.
• Assignment on choosing and applying an efficient sorting technique for a given problem.
• Assignment on comparison of different sorting techniques.
• Practical - Solving a given problem using efficient search technique.

Suggested Evaluation Methods:
• Quizzes on basics of hashing and sorting.
• Assignments on creation and manipulation of hash table, priority queues.
• Demonstration of the practical implementations.

TOTAL: 45 PERIODS
OUTCOMES:
On completion of the course, the students will be able to:
1. Demonstrate basic and advanced concepts of C programming language.
2. Use abstract data types including stacks, queues and lists for any application.
3. Design and implement tree data structures.
4. Analyze and implement hashing techniques that solve in linear time.
5. Apply sorting algorithms for a given problem.
6. Choose appropriate data structure and implement a given application.

REFERENCES:

<table>
<thead>
<tr>
<th></th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>CO2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>CO3</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>CO4</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>CO5</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>CO6</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>3</td>
</tr>
</tbody>
</table>

CA5202 OPERATING SYSTEMS

OBJECTIVES:
- To provide an understanding of the major operating system components.
- To describe the services an operating system provides to users, processes and other systems.
- To describe various features of processes including scheduling, creation and termination.
- To present both software and hardware solutions of the critical section problems.
- To explain the functions of file system and performance aspects of I/O hardware and software.
UNIT I  INTRODUCTION TO OPERATING SYSTEMS


Suggested Activities:
- Practical - Introduction to xv6: download, build, boot (in virtual machine if needed).
- Practical - Implementation of a user program in xv6 to print “Hello Welcome to shell Programming!!”.
- External learning - Explore the xv6 processes: fork(), exit(), wait(), kill(), exec(), sleep() and wakeup().
- Flipped classroom on asynchronous overlapping processes.

Suggested Evaluation Methods:
- Discussion and questionnaire on build and boot of xv6.
- Assessing the implemented program.
- Quiz on xv6 system calls and processes.
- Discussion and quiz on asynchronous overlapping processes.

UNIT II  THREADS AND CPU SCHEDULING


Suggested Activities:
- Study on how the system calls can be used to create kernel threads.
- Practical - Create thread and implement multi-threading using pthread library using any language.
- Practical - Study on xv6 scheduling policies and implement xv6 priority scheduling.
- Flipped classroom on scheduling mechanisms versus policies.

Suggested Evaluation Methods:
- Quiz to judge the understanding of threads.
- Assessing the implemented program.
- Quiz to check the understanding of the scheduling concepts in xv6.
- Discussions and assignment evaluation on scheduling mechanisms.

UNIT III  PROCESS SYNCHRONIZATION


Suggested Activities:
- Practical - Implementation of at least one form of producer consumer problem using any programming language.
- Practical - Implementation a mutex locks using any programming language.
- Practical - Implementation of counting semaphores in xv6.
Suggested Evaluation Methods:
- Evaluation of the implemented programs.

UNIT IV  MEMORY MANAGEMENT

Suggested Activities:
- Flipped classroom on various segmentation schemes.
- Analyze and justify why mobile operating systems such as android, iOS do not support swapping.
- Study on how memory management and paging works in xv6.
- Practical - Implementation of copy-on-write fork in xv6.

Suggested Evaluation Methods:
- Quiz on segmentation schemes.
- Discussions on swapping.
- Quiz on memory management and paging of xv6.
- Assessing the understanding of copy-on-write fork in xv6 through programming assessment.

UNIT V  I/O SYSTEMS
I/O Hardware – Application I/O Interface – Kernel I/O Subsystem – Communication with I/O devices – STREAMS.

Suggested Activities:
- External learning - Study on I/O system calls (open, read, write, ioctl, close) in xv6.
- Analyzing and identifying the issues to be addressed while assigning priorities to different interrupts, handling simultaneous interrupts from different devices.

Suggested Evaluation Methods:
- Classroom quiz on I/O system calls in xv6.
- Cooperative discussion on handling interrupts.

PRACTICAL EXERCISES:
1. Introduction to Linux.
2. Experiment most common system calls(e.g OPEN, CREAT, READ, WRITE, CLOSE, LSEEK)in order to make input output operations on files, as well as operations to handle files and directories in Linux.
3. Create, work with and manipulate processes in Linux.
4. Practical - Implement FCFS and SJF scheduling algorithm.
6. Practical - Implement a program using pthreads.
7. Practical - Implement a basic Semaphore.
8. Practical - Implement interprocess communication using pipes.
9. Practical - Implement interprocess communication using signals.
10. Practical - Implement a program to design a game using Unix pipes and IPC through shared memory and message queues.

TOTAL: 75 PERIODS
OUTCOMES:
On completion of the course, the students will be able to:
1. Describe how operating systems have evolved over time from primitive batch systems to sophisticated multi-user systems.
2. Understand the basic concepts of operating system process control, synchronization, and scheduling.
3. Understand the concepts and techniques involved in operating system memory management, secondary storage and file systems.
4. Explain the basic structure and functions of operating systems.
5. Identify the problems related to process management and synchronization and apply learned methods to solve basic problems.
6. Demonstrate knowledge in applying system software and tools available in modern operating systems.

REFERENCES:

<table>
<thead>
<tr>
<th></th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>CO2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>CO3</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>CO4</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>CO5</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>CO6</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>
OBJECTIVES:

- To understand the concepts of software processes, process models and fundamental process activities.
- To understand the fundamental concepts of requirements engineering & requirements specification and documents.
- To know about the idea of design patterns and how these are away of reusing design knowledge and experience.
- To be aware of testing processes, techniques and debugging to solve program defects.
- To learn how to use software metrics, manage risk, apply basic software quality assurance practices to ensure that software designs, development, and maintenance meet or exceed applicable standards.

UNIT 1 PROCESS


Suggested Activities:

- In-class activity - Application specific product and process view.

Suggested Evaluation Methods:

- Quizzes on different types of models.
- Assignments on selection of suitable software process models for a given software specification.
- Tutorials on identification of sample application for each process model and justification of the same stating reasons.

UNIT II SOFTWARE REQUIREMENTS


Suggested Activities:

- In class activity on software projects like an embedded computer system operating in real time. The following tasks may be performed:
  - Take a real time project and elicit requirements, and form a software requirements Specification document.
  - Draw a process model showing how requirements review might be organized.

Suggested Evaluation Methods:

- Tutorial on various requirements elicitation mechanisms and selection of an appropriate strategy.
- Assignment on requirements categorization (considering contradicting, omission, commission of requirements) in a software project.
UNIT III ANALYSIS AND DESIGN


Suggested Activities:
- External learning - Use open source tools to perform modeling approaches.
- In-class activity – Draw UML models for any given real time application.

Suggested Evaluation Methods:
- Assignment on determine the flow of data/events among the processes in the application under consideration.
- Assignment on designing UI of Sample application

UNIT IV SOFTWARE TESTING


Suggested Activities:
- External learning - Use open source testing tools to test the program defects and debug it.
- In-class activity on developing test cases for Equivalence class partitioning.
- In-class activity on developing test cases for Boundary value analysis.
- In-class activity on developing test cases for Basis Path testing.
- In-class activity on developing test cases for Control, structure testing.

Suggested Evaluation Methods:
- Assignment on testing of sample application.
- Assignment on testing sample application using Black box and White box approaches and understand the differences in selecting of test cases from the test suite.
- Case studies based on any real time application projects.

UNIT V MANAGEMENT AND METRICS


Suggested Activities:
- External learning - Tools for estimating software cost.
- External learning - Software Quality Models.
- In-class activity on FP metrics & Variants.

Suggested Evaluation Methods:
- Tutorial on Identification of potential risks for software project during development/maintenance and tabulate.
- Assignment on preparation of Software Configuration Management template for a software project.
- Calculation of Test metrics for Sample application.

TOTAL: 45 PERIODS
OUTCOMES:
On completion of the course, the students will be able to:
1. Understand the role and impact of software engineering in contemporary business, global, economic, environmental and societal context.
2. Elicit the requirements for real, time problems. Analyze and use open source tools for project designing.
3. Develop User Interface design for the given system.
4. Analyze and resolve information technology problems through the application of systematic approaches and diagnostic tools.
5. Estimate the cost of software and apply software management principles.
6. Understand the issue of Software Quality and activities present in typical Quality management process.

REFERENCES:

<table>
<thead>
<tr>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>CO2</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>CO3</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>CO4</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>CO5</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>CO6</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>3</td>
</tr>
</tbody>
</table>

OBJECTIVES:
- To learn the fundamentals of data modeling and design in advanced databases.
- To study the working principles of distributed databases.
- To have an introductory knowledge about the query processing in object-based databases and its usage.
- To understand the basics of spatial, temporal and mobile databases and their applications.
- To learn emerging databases such as XML, Data warehouse and NoSQL.

UNIT I DISTRIBUTED DATABASES
Suggested Activities:
- Practical - Design of distributed database with fragmentation using any DBMS.
- Flipped classroom on distributed transaction protocols.
- Writing distributed queries and optimizing the queries.

Suggested Evaluation Methods:
- Evaluation of designed Distributed Database system.
- Quizzes on distributed transactions.
- Tutorials on distributed queries and optimization.

UNIT II  NOSQL DATABASES

Suggested Activities:
- Exploring MongoDB using JAVA/Python/Ruby/PHP.
- Perform Database Operations using MongoDB/Cassandra/HIVE.
- Scenario based query development for database applications.

Suggested Evaluation Methods:
- Evaluation of the database operations.
- Tutorial on scenarios to analyze the need for DB in various applications.
- Quizzes on query language features.

UNIT III  ADVANCED DATABASE SYSTEMS

Suggested Activities:
- Individual/group activities for application specific data handling.
- Discussion about advantages and drawbacks of transaction models for different applications involving spatial-temporal data.

Suggested Evaluation Methods:
- Tutorials on active and deductive databases.
- Assignments on spatial databases.
- Quizzes on mobile database transactions.

UNIT IV  XML AND DATAWAREHOUSE

Suggested Activities:
- Flipped classroom on demonstrate the operations on XML data and data warehouse.
- Practical - Use tools to solve data access scenarios.
Suggested Evaluation Methods:
- Assignments on XML parsers, XSL and XQuery.
- Demonstration and presentation of the practical assignments.

UNIT V INFORMATION RETRIEVAL AND WEB SEARCH


Suggested Activities:
- Flipped classroom on queries in IR.
- Practical - Install any IR framework such as SOLR, and experiment with it.

Suggested Evaluation Methods:
- Practical demonstration on IR Queries.
- Quizzes on IR frameworks and related tools.

TOTAL: 45 PERIODS

OUTCOMES:
On completion of the course, the student will be able to:
1. Design a distributed database system and execute distributed queries.
2. Use NoSQL database systems and manipulate the data associated with it.
3. Have knowledge of advanced database system concepts.
4. Design a data warehouse system and apply OLAP operations.
5. Design XML database systems and validating with XML schema.
6. Apply knowledge of information retrieval concepts on web databases.

REFERENCES:

<table>
<thead>
<tr>
<th></th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>CO2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>CO3</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>CO4</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>CO5</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>CO6</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>
OBJECTIVES:
- To understand the basics of web standards and scripting languages.
- To design rich web pages using advanced scripts.
- To understand and develop simple PHP programs.
- To create dynamic web pages at server-side scripting with database connectivity.
- To develop MVC framework based web application using PHP.

UNIT I  BASICS OF WEB AND SCRIPTING  9

Suggested Activities:
- Flipped classroom on form design using HTML5 and CSS3.
- Practical - Design simple website with dynamic web pages.
- Practical - Array and regular expression using JavaScript.

Suggested Evaluation Methods:
- Quiz on topics related to design.
- Demonstration of the practical implementations.

UNIT II  ADVANCED SCRIPTING FEATURES  9

Suggested Activities:
- Flipped classroom on JavaScript with Jquery.
- External learning - Significance of AngularJS over JavaScript.

Suggested Evaluation Methods:
- Quizzes AngularJS.
- Assignments on Internet application using JSON.

UNIT III  INTRODUCTION TO PHP  9

Suggested Activities:
- Flipped classroom on XML and AJAX with PHP.
- External learning – Additional features in PHP.

Suggested Evaluation Methods:
- Quizzes on PHP.
- Assignments object features of PHP.

UNIT IV  SERVER – SIDE SCRIPTING  9
Suggested Activities:
- Practical - Implementation of simple web application using PHP.
- Practical - Implementation of database access using web application.

Suggested Evaluation Methods:
- Demonstrations of implementations.

UNIT V Advanced Frameworks


Suggested Activities:
- Practical - Installing the bootstrap.
- Practical - Writing simple application using Sparks like online shopping cart.

Suggested Evaluation Methods:
- Demonstration of application functions.

TOTAL: 45 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
1. Create simple web pages and incorporate client side validation.
2. Create dynamic websites with JQuery and AngularJS.
3. Write simple programs using PHP.
4. Develop web application using PHP and MySQL.
5. Develop web application using MVC architecture with advanced frameworks.
6. Develop a full-fledged web application incorporating advanced features.

REFERENCES:

<table>
<thead>
<tr>
<th>CO1</th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
</tbody>
</table>
OBJECTIVES:
- To introduce the concepts of structured programming language.
- To develop skills in design and implementation of data structures and their applications.
- To learn and implement linear data structures and nonlinear data structures.
- To study and implement hashing techniques.
- To study and analyze the different sorting and searching techniques.

EXPERIMENTS:
1. Implementation of simple programs in C using Data types, Variables, Conditional and Iterative Statements.
2. Implementation of simple programs in C using arrays and functions.
3. Implementation of simple programs in C using structures and unions.
5. Implementation of singly linked list ADT, doubly linked list ADT.
6. Implementation of circular linked list ADT and applications of lists.
7. Implementation of stack ADT using arrays and linked lists and applications of stack.
8. Implementation of queue ADT using arrays and linked lists.
9. Implementation of binary search tree ADT.
10. Implementation of hashing techniques such as separate chaining, open addressing.
11. Implementation of sorting algorithms – insertion sort, shell sort, merge sort.

TOTAL: 60 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
1. Apply knowledge to solve computer science and information technology problems using the basics of C programming and the concepts of data structures.
2. Choose and apply linear data structure for a given application.
3. Choose and apply non-linear data structures for a given application.
4. Apply different types of hashing techniques based on the problem requirements.
5. Use sorting techniques for a given real world application.
6. Use searching techniques to solve a given problem.
OBJECTIVES:
- To understand the basics of web standards and scripting languages.
- To design rich web pages using advanced scripts.
- To understand and develop simple PHP programs.
- To create dynamic web pages at server – side scripting with database connectivity.
- To develop MVC framework based web application using PHP.

EXPERIMENTS:
1. Write CSS rule that specifies the presentation of elements on a website to control its appearance.
2. Traverse, edit and modify elements in a HTML5 document using DOM.
3. Create a website with dynamic features using JavaScript event handling.
4. Create enhanced web pages using JQuery and AngularJS.
5. Write PHP programs to demonstrate the uses of array, string manipulation, file and exception handling.
6. Write PHP programs to connect to MySQL database and provide CRUD operations.
7. Develop simple PHP based web application for session handling.
8. Create an XML document; load the XML document into your web browser. Write an XML Schema document specifying the structure of the XML document created. Write an XSL style sheet for your solution that displays the content in an HTML5 table.
9. Develop PHP based internet rich application with AJAX features.
10. Develop PHP programs to demonstrate object features.
11. Create PHP based web application to attach files and send emails.
12. Create PHP application using Code Igniter framework to demonstrate MVC architecture.

TOTAL: 60 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
1. Create simple web pages and incorporate client side validation.
2. Create dynamic websites with JQuery and AngularJS.
3. Write simple programs using PHP.
4. Develop web application using PHP and MySQL.
5. Develop web application using MVC architecture with advanced frameworks.
6. Develop a full-fledged web application incorporating advanced features.
OBJECTIVES:

- To understand the network fundamentals.
- To explore various application layer protocols.
- To understand the transport layer services.
- To learn the network layer functionalities.
- To understand the link layer services and data communication fundamentals.

UNIT I  APPLICATION LAYER

Suggested Activities:

- Assignment on list of protocols in each layer of the TCP/IP standard.
- External learning - Socket programming.
- Assignment on classical problems on performance of applications based on delay, bandwidth metrics.

Suggested Evaluation Methods:

- Demonstration of HTTP/DNS format using Wireshark.
- Demonstration of Practical configuration of POP3.
- Quiz on working of BitTorrent.

UNIT II  TRANSPORT LAYER

Suggested Activities:

- Flipped classroom on UDP/ TCP packet segment.
- Flipped classroom on sliding window based flow control.
- External learning - Adaptive retransmissions.

Suggested Evaluation Methods:

- Quiz on TCP state transition.
- Demonstration of TCP/UDP segment format using Wireshark.

UNIT III  NETWORK LAYER

Suggested Activities:

- Flipped classroom on Virtual Circuit Switching.
- External learning - IP Packet format/ Fragmentation.
- External learning - Network Bridges.
- Assignment on classical problems on Subnetting/CIDR.
Suggested Evaluation Methods:
- Demonstration of analyzing IP/ARP/DHCP/ICMP packets using Wireshark.
- Quiz on BGP.
- Group discussion on IPv6.

UNIT IV DATA LINK LAYER


Suggested Activities:
- External learning - Bit/byte oriented protocols.
- Assignment on classical problems in checksum/CRC.
- Flipped classroom on sliding window protocol.

Suggested Evaluation Methods:
- Viva voce on assignment topics.
- Quiz on Ethernet/WiFi frame format.
- Demonstration on analyzing ethernet frames using Wireshark.
- Quiz on hubs/switches.

UNIT V FUNDAMENTALS OF DATA COMMUNICATION


Suggested Activities:
- Flipped classroom on transmission impairments.
- Assignment on classical problems in digital data and digital signal encoding.
- External learning - Channel capacity measurements.
- Assignment on differences between wired and wireless transmissions.

Suggested Evaluation Methods:
- Viva voce on assignment topics.
- Quiz on digital data and digital signal encoding.
- Quiz on transmission media types.

TOTAL: 60 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
1. Describe the fundamentals of internetworking.
2. Design new application layer protocols for various applications.
3. Select suitable transport layer protocols for network applications.
4. Trace and analyze the packets between end-to-end applications.
5. Calculate the capacity of links between nodes.
6. Identify suitable signal encoding techniques for various scenarios.

REFERENCES:
OBJECTIVES:
- To understand the object oriented concepts of Java.
- To learn GUI based application development and network programming.
- To build dynamic web sites using server side technologies with database connectivity.
- To learn the concepts of distributed objects, messaging and mail services.
- To understand the importance of advanced frameworks.

UNIT I JAVA BASICS

Suggested Activities:
- Flipped classrooms on basics of Java.
- Learning and Implementation in the following topics.
  - Create and manipulate character – string objects of class String, String Builder and Character.
  - Creating applications using system and user defined exceptions.

Suggested Evaluation Methods:
- Quiz on Java fundamentals.
- Demonstration of Java programs with object oriented features.

UNIT II GUI, I/O AND NETWORK PROGRAMMING

Suggested Activities:
- Learning and implementation in the following topics.
  - Java I/O Streams for text and binary data operations to read from and write to files.
  - Java Applications using Generics.
- Java Frame and Applet based Application Development.
- Java based thread implementation using thread priorities.
- Java networking applications using sockets and datagram's.

**Suggested Evaluation Methods:**
- Quiz on generics and networking.
- Tutorial assignments on advanced Java features.

**UNIT III**
**JDBC AND WEB APPLICATION DEVELOPMENT**
9

**Suggested Activities:**
- Developing a database application using JDBC.
- Creation of simple servlet based application.
- Creation of JSF application and managing sessions.

**Suggested Evaluation Methods:**
- Quizzes on database application using JDBC.
- Demonstration of web applications developed using servlets, JSP and JSF.

**UNIT IV**
**DISTRIBUTED OBJECTS**
9

**Suggested Activities:**
- Developing distributed applications using RMI and Java Bean.
- Development of synchronous and asynchronous Java based messaging services.
- Creation of a SOAP and RESTful based web services.

**Suggested Activities:**
- Quiz on RMI, XML and web services.
- Demonstration of RMI, XML and web services implementation.

**UNIT V:**
**ADVANCED FRAMEWORKS**
9

**Suggested Activities:**
- Flipped classroom on MVC Architecture

**Practical Learning:**
- Create a simple application using struts.
- Hibernate framework based O/R mapping.
- To create simple applications using Spring framework.
Suggested Evaluation Methods:
- Demonstration on Hibernate, Struts and Spring framework based application.

OUTCOMES:
On completion of the course, the students will be able to:
1. Practical - Implement object oriented concepts of Java programming.
2. Work with Generics, networking and GUI based application development.
3. Develop dynamic web applications with database connectivity using server side technologies.
4. Create distributed applications using RMI, Java Bean and web services.
5. Design and development of applications using advanced frameworks.
6. Understand the importance of advanced frameworks.

REFERENCES:

<table>
<thead>
<tr>
<th></th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>CO2</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>CO3</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>CO4</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>CO5</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>CO6</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
</tr>
</tbody>
</table>

CA5303 ADVANCED DATA STRUCTURES AND ALGORITHM DESIGN

OBJECTIVES:
- To learn the usage of hierarchical data structures.
- To familiarize with various heap structures.
- To understand the usage of graph algorithms.
- To understand the techniques involved in analyzing the efficiency of computer algorithms.
- To learn and understand the usage of algorithm design paradigms to solve real life problems.
UNIT I ADVANCED NON-LINEAR DATA STRUCTURES

Suggested Activities:
- Flipped classroom on binary search tree and tree traversals.
- External learning on 2-3 Trees, M-ary Trees.
- Practical - Choose and apply a suitable tree structure for solving a given real time problem such as organization of data points in K-dimensional space.

Suggested Evaluation Methods:
- Assignments on problem solving on 2-3 trees and M-ary trees.
- Quizzes on binary trees and tree traversals.

UNIT II HEAP STRUCTURES

Suggested Activities:
- Flipped classroom on binary heap operations.
- External learning on D-heaps.
- Practical - Implementation on min max heaps.

Suggested Evaluation Methods:
- Assignments on problem solving on Fibonacci heap operations using amortized analysis.
- Quizzes on heap operations and D-heaps.

UNIT III GRAPH ALGORITHMS

Suggested Activities:
- Flipped classroom on basics of graphs and graph operations.
- External learning on applications of graphs and DFS.
- Practical Learning to choose and apply a suitable graph algorithm for solving a real time problem/scenario such as finding shorter routes in networks, finding relationship in social network graphs.

Suggested Evaluation Methods:
- Assignments on applications of graphs and DFS.
- Quizzes on graph operations.
- Demonstration on practical learning.

UNIT IV ALGORITHMS IN COMPUTING
Suggested Activities:
- Flipped classroom on basics of algorithms and design of algorithms.
- Assignment on finding order of growth for exponent and logarithmic time algorithms.
- Assignment on analysis of time complexity for different algorithms such as sorting, searching and series generation.
- Assignment on solving recurrence relations using substitution and recursion tree method.
- Assignment on formulation of recurrence equations for recursive programs such as Tower of Hanoi, staircase, and triangular number problems.

Suggested Evaluation Methods:
- Assignments on problem solving exercises.
- Evaluation of order of growth for various algorithms.
- Evaluation of the assignments.
- Evaluation of recurrence relations solutions.

UNIT V ALGORITHM DESIGN TECHNIQUES


Suggested Activities:
- Flipped classroom on basics of algorithm design strategies.
- External learning - Backtracking algorithms, e.g., n queens problem.
- Practical - Choose and apply a suitable algorithm design technique for solving real time problems such as puzzle solving, checker board and job selection.

Suggested Evaluation Methods:
- Assignments on backtracking techniques.
- Quizzes on algorithm design strategies.
- Demonstration of practical learning.

TOTAL: 45 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
1. Apply hierarchical data structures based on the problem requirements.
2. Apply different heap data structures to solve real time problems.
3. Design algorithms using graph structures to solve real life problems.
4. Analyze and compare the algorithms based on their efficiency.
5. Solve real time problems by implementing learned algorithm design techniques and data structures.
6. Solve NP complete problems using appropriate methods.

REFERENCES
OBJECTIVES:
- To understand and apply the fundamentals of core Java.
- To implement inheritance, polymorphism, interfaces, multithreading, streaming, networking, generic collections and RMI.
- To develop web applications using client side and server side programming.
- To understand SOAP and REST based web service standards.
- To learn and use MVC architecture for application development.

EXPERIMENTS:
1. Design and Implement java programs that deals with the following:
   a. Classes and Objects and Interfaces.
   b. Exception Handling with user defined Exceptions.
   c. String Handling (String Class objects – string manipulation functions).
   d. Streaming (image file handling using byte streams – text file manipulation using character streams).
   e. Implementation of Thread Synchronization using any application.
   f. Reading and Writing Objects using Serialization.
   g. Creation of User Interfaces using SWING and graphic features.
   h. Creation and Manipulation of generic objects.
2. Java socket programming.
   b. Implementation of simple http client/server application.
   c. Simulation of DNS protocol.
3. Reading websites using URL class.
4. Implementation of any Information System using JDBC.
6. Database Connectivity using Java Bean.
7. Web Application development using Servlet, JSP and JSF.
8. Session Management and Implementation of Cookies using JSF.
9. Development of SOAP and REST based web services.
10. Development of Hibernate framework based application for O/R mapping.

TOTAL: 60 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
1. Write programs on advanced features of Java such as streaming, networking, multithreading and generics.
2. Design and develop GUI based components and animations.
3. Develop chat and file transfer applications.
4. Create JDBC based distributed applications using RMI and Java Beans.
5. Develop dynamic data driven websites using server side programming.
6. Create MVC applications using advanced frameworks.

<table>
<thead>
<tr>
<th></th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>CO2</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>CO3</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>CO4</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>CO5</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>CO6</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
</tbody>
</table>

CA5312 ADVANCED DATA STRUCTURES AND ALGORITHMS
LABORATORY

OBJECTIVES:
- To understand the usage of advanced tree structures.
- To familiarize the usage of heap structures.
- To learn the usage of graph data structures and spanning trees.
- To learn how to analyze the complexities of algorithms.
- To explore the various design strategies of algorithms.

EXPERIMENTS:
1. Implementation of AVL tree and its operations.
2. Implementation of Splay tree and its operations.
3. Implementation of red-black tree and its operations.
4. Implementation of basic heap and leftist heap operations.
5. Implementation of Fibonacci heap operations.
7. Implementation of a spanning tree for a given graph using Prim’s algorithm.
8. Implementation of shortest path algorithms such as Dijkstra’s and Floyd Warshall’s algorithm.
9. Implementation of iterative and recursive algorithms with its complexity analysis.
10. Implementation of merge sort algorithm analysis using divide and conquer approach.

TOTAL: 60 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
1. Implement basic and advanced data structures extensively.
2. Choose and apply suitable hierarchical data structures for real time problems.
3. Apply suitable heap data structures based on the problem requirements.
4. Design and apply algorithms using graph structures.
5. Design and implement iterative and recursive algorithms with minimum complexity.
6. Design and develop efficient algorithms by adopting suitable algorithm design strategies.

<table>
<thead>
<tr>
<th></th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>CO2</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>CO3</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>CO4</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>CO5</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>CO6</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>

CA5313

SOCIALLY RELEVANT PROJECT

OBJECTIVES:
- To identify socially relevant problems.
- To design solutions for socially relevant problems.
- To develop projects based on software design process.
- To implement solutions for societal valued projects using relevant state of the art technologies.
- To test the implemented project based on user needs and usefulness.

Students are expected to take up problems that would directly benefit the society and design and implement an IT based solution for the problem, based on the courses undertaken up to that semester. The domains of the problems may reach out to sectors like but not limited to Energy, Education, Material, Environment, Telecommunications, Defense, Healthcare, Entertainment and Agriculture. The societal value of the project is to be evaluated based on the need of the hour and request from stakeholders. The evaluation of the project would be
based on the usefulness of the problem statement, formulation of the problem, stakeholders need, and the usage statistics of the solution and the technical merit of the solution.

The project design, development and testing phases may be as per the following:

**REQUIREMENTS ENGINEERING PHASE:**
- Problem identification.
- Feasibility study of domain.
- Requirement elicitation and analysis.

**DESIGN PHASE:**
- Architectural design.
- UI design.
- Component Design.
- Database design.

**IMPLEMENTATION PHASE:**
- Coding in a suitable language using necessary platforms and tools.

**TESTING AND VALIDATION PHASE:**
- Component Testing.
- System Testing.
- Acceptance Testing.

**TOTAL: 30 PERIODS**

**OUTCOMES:**
On completion of the course, the students will be able to:
1. Analyze social problems and provide technical solutions.
2. Benefit the society by providing IT based solutions for social problems.
3. Design, develop and implement solutions for social problems.
4. Develop innovative technical solutions of social relevance.
5. Design, develop and implement standard solutions to social problems applying Software engineering methodologies.
6. Evaluate the solution based on usefulness, effectiveness and user satisfaction.

**REFERENCES:**
1. https://www.niti.gov.in/
2. https://www.sih.gov.in/

<table>
<thead>
<tr>
<th></th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>CO2</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>CO3</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>CO4</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>CO5</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>CO6</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>3</td>
</tr>
</tbody>
</table>
OBJECTIVES:
- To know the fundamental concepts of data science and analytics.
- To learn fundamental data analysis using R.
- To understand various data modeling techniques.
- To learn the basic and advanced features of open source big data tools and frameworks.
- To study various analytics on stream data.

UNIT I   INTRODUCTION TO DATA SCIENCE AND BIG DATA  9

Suggested Activities:
- Case studies on big data application domain.
- Solving numerical problems in sampling, hypothesis testing.
- Converting real time decision making problems into hypothesis.

Suggested Evaluation Methods:
- Assignments on hypothesis testing.
- Group presentation on big data applications with societal need.
- Quizzes on sampling and statistical testing.

UNIT II   DATA ANALYSIS USING R  9

Suggested Activities:
- Practical - Perform univariate analysis on UCI datasets.
- Solve numerical problems in correlation and regression using sample real time data.
- Practical - Implement univariate, bivariate and multivariate analysis using R Studio.
- Given a data set, explore the features using data analysis in R.

Suggested Evaluation Methods:
- Assignment on univariate, bivariate and multivariate analysis.
- Demonstrate implementation of univariate, bivariate and multivariate analysis using R Studio.
- Assignment on comparative analysis of the two or more data sets using their features.

UNIT III   DATA MODELING  9
Suggested Activities:
• Practical - Implementation of Bayesian modeling using Weka tool.
• Practical - Given a data set, apply Bayesian and neural models using open source data modeling tools.
• Solve numerical problems on Eigen Value, Eigen Vector, etc. to understand the working principles of mining techniques.
• Demonstration on data distribution in HBase and MongoDB.

Suggested Evaluation Methods:
• Implementation demonstration of Bayesian modeling and other simple data preprocessing tasks using Weka tool.
• Implementation demonstration of practical exercises.
• Mini project (individual) - Given a data set and decision making scenario identify suitable modeling technique(s) and implement using Weka tool.

UNIT IV    DATA ANALYTICAL FRAMEWORKS


Suggested Activities:
• Case studies on MapReduce for text mining and simple linear problems using numerical methods.
• Design and develop algorithms to be executed in MapReduce involving numerical methods for analytics.

Suggested Evaluation Methods:
• Mini Project (Group) - Real time data collection, saving in Hive, implement analytical techniques using MapReduce tasks and result projection.

UNIT V    STREAM ANALYTICS


Suggested Activities:
• Case studies on the usage of stream analytics in popular search engines.
• External learning - Real time sentiment analysis, stock market predictions.
• Assignments on solving simple numerical problems involving moments and skewness.

Suggested Evaluation Methods:
• Assignment on the following: given a problem scenario identify suitable stream analytical technique(s).
• Quiz on all topics covered in stream analytics.

OUTCOMES:
On completion of the course, the students will be able to:
1. Convert real world problems to hypothesis and perform statistical testing.
2. Perform data analysis using R.
3. Work with big data platform and its analysis techniques.
4. Identify and design efficient modeling of very large data.
5. Implement suitable data analysis for stream data.
6. Write efficient MapReduce programs for small problem solving methods.
REFERENCES:

<table>
<thead>
<tr>
<th></th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>CO2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>CO3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>CO4</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>CO5</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>CO6</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>

CA5402 EMBEDDED SYSTEMS AND INTERNET OF THINGS

OBJECTIVES:
- To learn the internal architecture and programming of an embedded processor.
- To introduce interfacing I/O devices to the processor and to run, debug programs in an IDE.
- To build a small low cost embedded system using Open Hardware Platforms.
- To apply the concept of Internet of Things in real world scenario.
- To deploy IoT application and connect to the cloud.

UNIT I 8-BIT EMBEDDED PROCESSOR

Suggested Activities:
- Flipped classroom on activity on different types of microcontrollers.
- Assignment on writing simple assembly codes.
- Practical - Developing simple application using assembly code.

Suggested Evaluation Methods:
- Tutorials on Instruction Set and programming.
- Assignments on programming using machine code.
- Quizzes on interrupt handling.
UNIT II: EMBEDDED C PROGRAMMING


Suggested Activities:
- Flipped classroom on activity on different types of RTOS.
- Assignment on writing simple embedded C codes.
- Developing simple application using embedded C code.

Suggested Evaluation Methods:
- Demonstration of application using Embedded C programming.
- Assignment on scheduling policies.
- Quizzes on various topics of the unit.

UNIT III: FUNDAMENTALS OF IOT


Suggested Activities:
- Flipped classroom on protocols like Bluetooth, WiFi, ZigBee etc. standards.
- Case study of different sensors used in IoTs.

Suggested Evaluation Methods:
- Quizzes on IoT basics and levels.
- Assessment on IoT protocols.
- Assessment in finding the IoT levels for different applications.

UNIT IV: BUILDING IOT

Open Hardware Platforms: Interfaces, Programming, APIs and Hacks – Web Services – Integration of Sensors and Actuators with Arduino/ Raspberry Pi/ Other Light Weight Boards.

Suggested Activities:
- Developing simple application like Blinking led, controlling led using Arduino / Raspberry Pi.
- External learning on pcDuino, Beaglebone Black, Cubieboard.

Suggested Evaluation Methods:
- Tutorial problems on Arduino sketches.
- Assignment on Interfacing I/O based applications with Arduino/Raspberry Pi.
- Quizzes on web services.

UNIT V: APPLICATIONS


Suggested Activities:
- Flipped classroom on different existing IoT applications.
- Case study on Weather Monitoring System.
- Assignment on Analysis of data gathered from weather monitoring system.
- Collect and Process the simple IoT application data using XivelyPaaS.
Suggested Evaluation Methods:
- Assignment on different IoT based smart solutions.
- Quizzes on different types of storages and communications.

TOTAL: 45 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
1. Analyze architecture of embedded processors and micro controllers.
2. Design and deploy timers and interrupts.
3. Design and develop the prototype of embedded systems.
4. Familiarize with fundamentals of IoT.
6. Analyze and develop applications of IoT in real time scenario.

REFERENCES:

<table>
<thead>
<tr>
<th></th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>CO2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>CO3</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>CO4</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>CO5</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>CO6</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
</tbody>
</table>

CA5403 CLOUD COMPUTING L T P C 3 0 0 3

OBJECTIVES:
- To learn about the concepts of distributed systems.
- To understand distributed resource management.
- To study the basics of cloud computing.
- To study about virtualization and cloud resource management.
- To be aware of different Cloud platforms.
UNIT I INTRODUCTION TO DISTRIBUTED SYSTEM CONCEPTS

Suggested Activities:
- Practical - Implement RPC and Bankers algorithm.
- Create and distribute a Torrent file to share a file in LAN Environment.

Suggested Evaluation Methods:
- Demonstration and assessment of the implemented algorithms.

UNIT II INTRODUCTION TO CLOUD COMPUTING

Suggested Activities:
- Use Google collaboration tools: Create Google Docs, Sheets and Slides and share it with other users.
- Explore public cloud services like Amazon, Google, Sales Force, Digital Ocean etc.

SUGGESTED EVALUATION METHODS
- Quizzes on different service models and deployment models.
- Report submission - Comparison of various services provided by different Cloud Service Providers (configuration of VM, cost, network bandwidth etc.).

UNIT III CLOUD ENABLING TECHNOLOGIES

Suggested Activities:
- Create a simple web service using Python Flask/Java/any language [Web Service: Client-server model should be implemented using socket/http].
- Install Oracle Virtual Box/VMware Workstation and create a chat application [Note: Launch two virtual machines for chat application].

Suggested Evaluation Methods:
- Review web services implementation - Proper Connection should be established between the client and server to make use of the service offered by the Server.
- Review the working of application in virtual environment.

UNIT IV CLOUD MANAGEMENT, STORAGE AND SECURITY
Suggested Activities:
- Use security tools like ACUNETIX, ETTERCAP to scan web applications on the cloud, cloud networks for finding vulnerabilities, verifying leakage of information to an unauthorized third party.

Suggested Evaluation Methods:
- Report submission - Generate a detailed report describing vulnerabilities along with the suitable action that can be taken to remedy the loopholes.

UNIT V CLOUD SOFTWARE AND COMPUTING PLATFORMS
7
HDFS – MapReduce – Google App Engine (GAE) – Programming Environment for GAE – Architecture of GFS – Case Studies: Openstack, Heroku and Docker Containers – Amazon EC2, AWS, Microsoft Azure, Google Compute Engine.

Suggested Activities:
- Install and configure OpenStack all-in-one using Devstack/Packstack and Launch VMs in OpenStack through dashboard.

Suggested Evaluation Methods:
- OpenStack Dashboard should be accessed though web browser. Verify the working of instance by logging into it/pinging the instance.

TOTAL: 45 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
1. Appreciate distributed computing, distributed resource management.
2. Articulate the main concepts, key technologies, strengths and limitations of cloud computing.
3. Develop the ability to understand and use the architecture of compute and storage cloud, service and delivery models.
4. Explain the core issues of cloud computing such as resource management and security.
5. Choose the appropriate technologies, algorithms and approaches for implementation and use of cloud.
6. Establish own cloud environment using Openstack and work on it.

REFERENCES:
OBJECTIVES:
- To provide hands-on cloud and data analytics frameworks and tools.
- To use the Python/R packages for performing analytics.
- To learn using analytical tools for real world problems.
- To familiarize the usage of distributed frameworks for handling voluminous data.
- To write and deploy analytical algorithms as MapReduce tasks.

EXPERIMENTS:
Do the following experiments using R/Python:
1. Download, install and explore the features of R/Python for data analytics.
2. Use the Diabetes data set from UCI and Pima Indians Diabetes data set for performing the following:
   b. Bivariate Analysis: Linear and logistic regression modeling.
   c. Multiple Regression Analysis
   d. Also compare the results of the above analysis for the two data sets.
3. Apply Bayesian and SVM techniques on Iris and Diabetes data set.
4. Apply and explore various plotting functions on UCI data sets.

Implement the following using Hadoop, Map Reduce, HDFS, Hive:
1. Perform setting up and Installing Hadoop in its two operating modes: pseudo distributed and fully distributed.
2. Implement the following file management tasks in Hadoop: adding files and directories, Retrieving files and Deleting files
3. (i) Performing a MapReduce Job for word search count (look for specific keywords in a file)
   (ii) Implement stop word elimination problem: Input a large textual file containing one sentence per line and a small file containing a set of stop words (one stop word per line) and save the results in an output textual file containing the same sentences of the large input file without the words appearing in the small file.
4. Implement a MapReduce program that processes a weather data set to:
   (i) Find average, max and min temperature for each year in National Climate Data Centre data set.
(ii) Filter the readings of a set based on value of the measurement. The program must save the line of input files associated with a temperature value greater than 30.0 and store it in a separate file.

5. Install, deploy & configure Apache Spark cluster. Run Apache Spark applications using Scala.

6. Install and run Hive then use Hive to create, alter, and drop databases, tables, views, functions, and indexes.

7. Mini projects on the following:
   (i) Simulate a simple recommender system with Amazon product dataset, Social tweet data set etc. on Hadoop.
   (ii) Perform a very large text classification run on Hadoop.

TOTAL: 60 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
1. Install analytical tools and configure distributed file system.
2. Have skills in developing and executing analytical procedures in various distributed frameworks and databases.
3. Develop, implement and deploy simple applications on very large datasets.
4. Implement simple to complex data modeling in NoSQL databases.
5. Develop and deploy simple applications in cloud.
6. Implement real world applications by using suitable analytical framework and tools.

<table>
<thead>
<tr>
<th></th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>CO2</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>CO3</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>CO4</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>CO5</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>CO6</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>1</td>
</tr>
</tbody>
</table>

CA5412   INTERNET OF THINGS AND CLOUD LABORATORY   L T P C
0 0 4 2

OBJECTIVES:
- To learn tools relevant to embedded system and IoT development.
- To write simple assembly programs that uses various features of the processor.
- To design and develop IoT application Arduino/Raspberry pi for real world scenario.
- To develop web applications in the cloud.
- To learn the design and development process involved in creating a cloud based application.
EXPERIMENTS:

PART I:
1. Implement assembly and Interfacing Programs Using Embedded C.
2. Embedded Application Development
   (i) Using Arduino and Raspberry Pi
   (ii) Using Bluemix platform
3. IoT Application Development
   (i) Using sensors and actuators (temperature sensor, light sensor, infrared sensor)
   (ii) Interfacing sensors with Arduino/Raspberry Pi/other equivalent boards
   (iii) Reading data from sensors
4. Explore different communication methods with IoT devices.
5. Collecting and processing data from IoT systems in the cloud using XivelyPaaS.

PART II:
7. Create a VM image which has a C Compiler along with an operating system and do the following experiments
   (i) Fibonacci series.
   (ii) File operations.
8. Install Virtual box with different flavors of Linux or Windows OS on top of windows 7 or 8.
9. Install GAE and run a quick sort using Python.
10. Install and run and run Eucalyptus to setup the network bridge.
11. Create two nodes in Eucalyptus and exchange data.
12. Mini Project: Simulate a cloud scenario using CloudSim and create, install and run a scheduling algorithm not present in CloudSim.

TOTAL: 60 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
1. Write and implement simple assembly programs that use various features of the processor.
2. Test and experiment different sensors for application development Arduino/Raspberry Pi/Equivalent boards.
3. Develop IOT applications with different platform and frameworks.
4. Become familiar with the basics of cloud computing.
5. Design and develop highly scalable cloud – based applications by creating and configuring virtual machines on the cloud and building private cloud.
6. Attempt to generate new ideas and innovations in cloud computing.

<table>
<thead>
<tr>
<th>CO1</th>
<th>CO2</th>
<th>CO3</th>
<th>CO4</th>
<th>CO5</th>
<th>CO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
</tr>
</tbody>
</table>
CA 5413

OBJECTIVES:
- To analyze and design an application using OOAD methodology
- To learn about the use of UML diagrams
- To be familiar with web services and micro services
- To be familiar with the use of web services for developing SOA based application
- To understand the basics of agile methodology and open source tools for development.

EXPERIMENTS:
1. Draw use case, class, and sequence diagram using Agro UML / Rational Software and realize the OO solution with automatic code generation.
2. Draw state diagram and other UML diagram for object oriented design and deployment.
3. Create web services using Java and use these web services in enterprise application development using XML standards.
4. Create web services using Python / .NET framework and use these web services in enterprise application development.
5. Develop RESTful web services and explore its uses in IoT environment.
6. Develop micro service and consume it using appropriate framework.
7. Explore the open source framework available for Agile development for continuous integration and continuous delivery.
8. Explore the features of Docker container and Kubernetes container orchestration platform and also study configuration management and monitoring features.

TOTAL: 30 PERIODS

OUTCOMES:
On completion of the course, the student will be able:
1. Analyze and design applications using Object Oriented methodologies.
2. Draw UML diagrams and understand its significance in OO software development.
3. Develop microservice and consume it using appropriate framework.
4. Develop web services and use them in enterprise level application.
5. Realize SOA based solution for smart home / IoT enabled environment.
6. Use devops tools and be familiar with the significance of them in software environment.

CA5501

OBJECTIVES:
- To understand the need and characteristics of mobile applications.
- To design the right user interface for mobile application.
- To understand the design issues in the development of mobile applications.
- To understand the development procedure for mobile application.
- To develop mobile applications using various tools and platforms.

UNIT I INTRODUCTION
Suggested Activities:
- Flipped classroom on survey on mobile application models.
- External learning - Mobile application design using frameworks and tools.

Suggested Evaluation Methods:
- Quizzes related to mobile application models.
- Assignment on mobile application design using frameworks and tools.

UNIT II USER INTERFACE
Generic UI Development – Designing the Right UI – Multimodal and Multichannel UI – Gesture Based UI – Screen Elements and Layouts – Voice XML.

Suggested Activities:
- Discussion on UI for mobile application like voice and gestures.
- External learning - Survey of different view elements for mobile application.

Suggested Evaluation Methods:
- Quiz on user interface design for mobile applications.
- Assignment on different view elements for mobile application.

UNIT III APPLICATION DESIGN

Suggested Activities:
- Discussion on memory constraints for mobile application design
- External learning - Survey of resource management and concurrent operations.

Suggested Evaluation methods:
- Quiz on memory constraints in design for mobile applications
- Assignment on content management system like Moodle.

UNIT IV APPLICATION DEVELOPMENT I

Suggested Activities:
- Simple Android application development like user account creation.
- Developing Android application for accessing the mobile database to view user data.

Suggested Evaluation Methods:
- Demonstration of application functionality using emulators.

UNIT V APPLICATION DEVELOPMENT II

Suggested Activities:
- Developing web application.
- Practical - Android application accessing GPS for location based service.
Suggested Evaluation Methods:
- Demonstration of web application.
- Demonstration of android application accessing GPS for location based service.

TOTAL: 45 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
1. Design the right user interface for mobile application.
2. Implement mobile application using UI toolkits and frameworks.
3. Design a mobile application that is aware of the resource constraints of mobile devices.
4. Develop web based mobile application that accesses internet and location data.
5. Implement android application to use telephony for SMS communication.
6. Implement android application with multimedia support.

REFERENCES:

<table>
<thead>
<tr>
<th>CO1</th>
<th>PO1</th>
<th>CO2</th>
<th>PO2</th>
<th>CO3</th>
<th>PO3</th>
<th>CO4</th>
<th>PO4</th>
<th>CO5</th>
<th>PO5</th>
<th>CO6</th>
<th>PO6</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>2</td>
<td></td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td></td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td></td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td></td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td></td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td></td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

CA5502 INFORMATION SECURITY

OBJECTIVES:
- To introduce the concepts and models of security.
- To understand the risk assessment and security standard.
- To plan for business continuity and incident response plan.
- To estimate the level of security risk faced by an organization and the counter measures to handle the risk.
- To understand potential vulnerabilities and to develop a security blueprint.

UNIT I INTRODUCTION TO SECURITY & SECURITY MODELS
Suggested Activities:
- In class activity to learn about various security services and attacks.
- In class activity to understand importance of various security models.
- External learning - Virus programs to demonstrate the virus attack.

Suggested Evaluation Methods:
- Assignment on SecSDLC to understand about the importance of each models.
- Quiz on Security attacks and services.

UNIT II SECURITY ANALYSIS AND LOGICAL DESIGN
9

Suggested Activities:
- Design security architecture and assess the risk in web application.
- Analysis risk for any real time applications and prepare blueprint for security to controlling the risk.
- Case study of various existing ISO standard security policies.

Suggested Evaluation Methods:
- Assignment on security architecture, DoD and security perimeter.
- Quiz on security polices and ISO standrads.

UNIT III PLANNING FOR CONTINUITY
9

Suggested Activities:
- Develop an attack success scenario and assess the potential damage.
- Prepare the contingency planning documents for business continuity.
- Study about the benefits and drawback of Law Enforcement Involvement.

Suggested Evaluation Methods:
- Assignment on disaster recovery planning and business continuity.
- Quiz on incident response, reaction and recovery.

UNIT IV SECURITY ANALYSIS
9

Suggested Activities:
- Highlight different security technology and its applications.
- Discussion on scanning and analysis tools for identify the vulnerabilities.
- Prepare security auditing report of an application and understanding the vulnerabilities in the system.
Suggested Evaluation Methods:
- Assignment to learn about vulnerability, analysis flaw hypothesis methodology, NRL taxonomy and Aslam’s model.
- Quiz on intruders, malicious software, firewalls, scanning and analysis tools.

UNIT V SECURITY PRACTICES

Suggested Activities:
- Use various scanning tools and gather information about vulnerable applications.
- Simulation of the Damn Vulnerable Web application to demonstrate various attacks.
- Practical - Implement cross-side scripting XSS and SQL injection in the web and database application.

Suggested Evaluation Methods:
- Assignment to understand OWASP/SANS top vulnerabilities and identify various attacks.
- Quiz on database security and social engineering.
- Demonstrate the tool to analyze various attacks like buffer overflow, XSS etc.

TOTAL: 45 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
1. Apply the basic security models and policies required by computing systems.
2. Develop a secure application using cryptographic algorithm.
3. Able to provide the security law and policies for an organization.
4. Predict the vulnerabilities across any computing system and hence be able to design a security solution for any computing system.
5. Understand the importance of security audit and risk management of an organization.
6. Able to understand various OWASP/SANS top vulnerabilities and perform penetration testing and security measures in a given application.

REFERENCES:

CA5511 MOBILE APPLICATION DEVELOPMENT LABORATORY

OBJECTIVES:
- To understand the need and characteristics of mobile applications.
- To design the right user interface for mobile application.
- To understand the design issues in the development of mobile applications.
- To understand the development procedure for mobile application.
- To develop mobile applications using various tools and platforms.
EXPERIMENTS:
1. Develop an application that uses GUI components, font and colours.
2. Design an application that uses Layout Managers and event listeners.
3. Develop a native calculator application.
4. Design an application that draws basic graphical primitives on the screen.
5. Develop an application that makes use of mobile database.
6. Develop an application that makes use of internet for communication.
7. Implement an android application that writes data into the SD card.
8. Implement an application that creates an alert upon receiving SMS message.
9. Develop a native application that uses GPS location information.
10. Develop a mobile application that creates a notification as task reminder.
11. Develop an android application using telephony to send SMS.
12. Implement primitive graphics in android application for color fill in objects.

OUTCOMES:
On completion of the course, the students will be able to:
1. Design the right user interface for mobile application.
2. Implement mobile application using UI toolkits and frameworks.
3. Design mobile applications that are aware of the resource constraints of mobile devices.
4. Develop web based mobile application that accesses internet and location data.
5. Implement android application to use telephony for SMS communication.
6. Implement android application with multimedia support.

<table>
<thead>
<tr>
<th></th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>CO2</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>CO3</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>CO4</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>CO5</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>CO6</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>3</td>
</tr>
</tbody>
</table>

CA5512  SOFTWARE DEVELOPMENT LABORATORY  L T P C
0 0 4 2

OBJECTIVES:
- To acquire the generic software development skill through various stages of software life cycle.
- To explore the methods and tools of software development.
- To generate test cases for software testing.
- To ensure the quality of software through software development with various environment.
- To practice software engineering techniques to system development and apply appropriate metrics.
EXPERIMENTS:
The lab can adhere to Pair Programming technique. If necessary distributed environment (like Saros) for enable pair programming can be incorporated.

1. A) Identifying the main deliverables of a project by creating a perfect Work breakdown structure (WBS) for the project chosen by the team. Use Gannt chart or any suitable graphical tool to estimate the project completion time and cost.
   B) Interpret the WBS of the project and find the dependency among the modules. Schedule the modules/activities as per the expertise of the team mates with the consideration to the dependency among the modules.
2. Analyze the possible risk(s) for each module under study by performing the risk assessment and decide on suitable contingency measures for high priority risks.
3. Usage of Version control software to handle the change in requirements, mapping of the requirements to WBS and check for traceability.
4. A) Construct the User Stories for each module to understand the need of the module for any end user. Indicate any 5 (minimum) functional and non – functional requirements.
   B) Consolidate the user stories to – similar, contradicting, etc. and select the set of requirements that could be completed within the scheduled time and budget.
   C) Construct a Use case Diagram considering the prioritized requirements. Use proper use cases and actors in the diagram. Use standard – UML Notations
5. Draw a UML sequence diagrams for each use case with at least 5 roles and 10 messages describing an entire process. The messages can be on a high level.
6. Draw a UML Class diagram including all necessary classes, associations, generalizations and attributes. Also identify the main for each class, its scope and arguments.
7. Mapping of classes identified in the Class Diagram to code in an appropriate choice of language to generate the skeleton of the project.
8. Implement the modules in a programming language of your choice with appropriate GUI and controls.
9. Use appropriate tools to debug the software during implementation. Usage of breaking points, inspecting values, restarting certain functions etc. to identifying the bug.
10. Write appropriate test cases for the modules that has been implemented
11. Perform the various levels of testing – Unit, Integration, System and Regression testing
12. Perform black box testing using equivalence class partitioning and boundary value analysis to find latent errors. Analyze why the need for test case is minimized using the approach.
13. Perform white box testing like code, branch and path coverage to eliminate dead code in the project.
14. Usage of appropriate quality metrics to identify the performance of the software as per the CMM level assumed.

TOTAL: 60 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
1. Apply software engineering practices and tools to the development of significant software components and systems.
2. Work within a defined software process and to contribute actively to improve the system.
3. Work in a team and to contribute to the overall success of a software development organization.
4. Plan and track project activities.
5. Communicate project and process information in written and oral form.
6. Analyze and apply independently learned knowledge and skills to the development of software components and systems.
OBJECTIVES:
- To decompose a blockchain system’s fundamental components, how they fit together and examine a decentralization using blockchain.
- To explain how Cryptocurrency works, from when a transaction is created to when it is considered part of the blockchain.
- To explain the components of Ethereum and programming languages for Ethereum.
- To study the basics Hyperledger and Web3.
- To provide details of alternative blockchain and blockchain projects in different perspective.

UNIT I INTRODUCTION TO BLOCKCHAIN

Suggested Activities:
- External learning - Programming to create your own Blockchain.
- Flipped classroom on studying Blockchain security issues.

Suggested Evaluation Methods:
- Practical assessment to be conducted to evaluate the program for creating Blockchain.

UNIT II INTRODUCTION TO CRYPTOCURRENCY

Suggested Activities:
- External learning - Creating the Wallets.
- Flipped classroom on showing the tracking process of transactions in Cryptocurrency.
Suggested Evaluation Methods:
- Assignment to be given on cryptocurrency failures.

UNIT III  ETHEREUM

Suggested Activities:
- External learning - Exploring Ethereum tools like Ganache and GO.
- Practical - Setup the Ethereum development environment.
- Practical - Develop smart contract on private Blockchain.

Suggested Evaluation Methods:
- Evaluation of developed smart contract on private Blockchain

UNIT IV  WEB3 AND HYPERLEDGER

Suggested Activities:
- Practical - Creating and deploying a business network on Hyperledger Composer Playground.
- Practical - Implementation of business network in Blockchain using hyperledger Fabric.

Suggested Evaluation Methods:
- Evaluation of developed business network on hyperledger fabric.

UNIT V  ALTERNATIVE BLOCKCHAINS AND NEXT EMERGING TRENDS

Suggested Activities:
- External learning - Blockchain using multichain.
- Assignments on Blockchain frameworks and business applications.

Suggested Evaluation Methods:
- Practical assessment of developing Blockchain based solution using Multichain for banking system.

OUTCOMES:
On completion of the course, the students will be able to:
1. Understand the technology components of Blockchain and how it works behind – the scenes.
2. Be aware of different approaches to developing decentralized applications.
3. Understand the Bitcoin and its limitations by comparing with other alternative coins.
4. Establish deep understanding of the Ethereum model, its consensus model and code execution.
5. Understand the architectural components of a Hyperledger and its development framework.
REFERENCES:

<table>
<thead>
<tr>
<th></th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>CO2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>CO3</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>CO4</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>CO5</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>CO6</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>3</td>
</tr>
</tbody>
</table>

CA5002 ETHICAL HACKING L T P C 3 0 0 3

OBJECTIVES:
- To explore the concepts of security testing and the knowledge required to protect against the hacker and attackers.
- To understand reconnaissance and the publicly available tools used to gather information on potential targets.
- To discover the scanning techniques used to identify network systems open ports.
- To identify network system vulnerabilities and confirm their exploitability.
- To explore techniques for identifying web application vulnerabilities and attacks.

UNIT I INTRODUCTION TO HACKING

Suggested Activities:
- In-class activity to understand the penetration testing methodologies.
- Practical - Use security tools in Kali Linux to assess the vulnerabilities.
- Prepare Vulnerability Assessment summary reports.
Suggested Evaluation Methods:
- Assignment on categories of penetration testing and vulnerability summary reports.
- Quiz on penetration testing methodologies, OSSTMM and OWASP.

UNIT II INFORMATION GATHERING AND SCANNING

Suggested Activities:
- Explain different ways to gather the information of a system in the network.
- Demonstrate the network command tools to identify the system.
- Understand the network protocols and port scanning techniques using Kali Linux.

Suggested Evaluation Methods:
- Assignment problems on information gathering and traceroute of ICMP, DNS and SNMP.
- Quizzes on enumeration, port scanning techniques and firewall/IDS evading techniques.

UNIT III NETWORK ATTACKS

Suggested Activities:
- Familiarizing with different types of attacks such as sniffing, spoofing etc.
- Demonstrating the MITM attack using ARP Poisoning using Kali Linux.
- Teaching with case studies: SSL Stripping, SQL Injection, Brute Force attacks.

Suggested Evaluation Methods:
- Assignment on denial of service (DoS) attack and hijacking session with MITM attack.
- Quizzes on SSL stripping, ARP spoofing and weak authentication.

UNIT IV EXPLOITATION
Suggested Activities:
- Case studies: Understand the Metasploit and Exploitations.
- Demonstrating email with malicious attachment and cracking the hashes.
- Practical - Implementing hashing algorithms and cracking the hashes.

Suggested Evaluation Methods:
- Assignments on social engineering toolkit and browser exploitation.
- Quizzes on reconnaissance with Metasploit and client–side exploitation methods.

UNIT V           WIRELESS AND WEB HACKING


Suggested Activities:
- Cracking the WEP and WPA/WPA2 passphrase using Cracking tool in Kali Linux.
- Design a web application with different authentication mechanism.
- Understand the protection mechanism to prevent against various server attacks.

Suggested Evaluation Methods:
- Assignment on evil twin attack and denial of service attack on access point in WLAN.
- Quizzes on types of authentication and vulnerabilities in a web application.

TOTAL: 45 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
1. Use the various security tools to assess the computing system.
2. Predict the vulnerabilities across any computing system using penetration testing.
3. Identify prediction mechanism to prevent any kind of attacks.
4. Protect the system from malicious software and worms.
5. Evaluate the wireless network flaws and able to apply security patches.
6. Analyze the risk and support the organization for effective security measures.

REFERENCES:

<table>
<thead>
<tr>
<th></th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>CO2</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>CO3</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>CO4</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>CO5</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>CO6</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
</tbody>
</table>
OBJECTIVES:
- To introduce big data, its evolution and applications.
- To familiarize the students with fundamental data analysis using R.
- To expose the students to different big data frameworks with R.
- To learn about integrating R and Hadoop.
- To learn about machine learning methods in RStudio.

UNIT I  INTRODUCTION TO R


Suggested Activities:
- Survey of R features for data analytics.
- Case studies on R equivalent features in other open source analytical tools.
- Remembering activities for R commands.

Suggested Evaluation Methods:
- Programming assignments to basic R objects and operations.
- Assignments on classification and summarization of various commands in R.
- Quiz basic R commands.

UNIT II  DATA ANALYTICS USING R


Suggested Activities:
- Exercises on aggregate functions in R.
- Solving numerical problems in sampling, hypothesis testing – t – test and ANOVA.
- Converting real time decision making problems into hypothesis.

Suggested Evaluation Methods:
- Student assignment on problem formation, hypothesis testing in R.
- Simple Lab tasks to apply visualization commands on standard data sets.
- Lab quiz on visualization commands in R.

UNIT III  R WITH NOSQL DATABASES


Suggested Activities:
- Programming Exercises on importing different types of data using R.
- Installation and configuring MongoDB.
- Trialing data importing using mongolite package.
Suggested Evaluation Methods:
- Student assignment exploring different data types in R.

UNIT IV INTEGRATING R AND HADOOP


Suggested Activities:
- Demonstration of Installation and configuration of Hadoop and R in Hadoop.
- Demonstration on simple sorting, searching application in Hadoop.

Suggested Evaluation Methods:
- Mini projects about word search from large text files in Hadoop.

UNIT V MODELING WITH R

Machine Learning methods in R – Naïve Bayes with H2O on Hadoop with R: Running an H2O instance – Reading and exploring the data in H2O – Naïve Bayes on H2O with R – Neural Networks with H2O on Hadoop with R.

Suggested Activities:
- Demonstration of Bayesian, neural network based data modeling using small datasets.
- Demonstration on programs to read, write and visualize data in H2O.
- Survey of other data modeling features in H2O.

Suggested Evaluation Methods:
- Mini projects involving data handling using H2O.
- Lab exercises to read different data from heterogeneous sources into H2O.

TOTAL: 45 PERIODS

OUTCOMES:
On completion of the course, the student will be able to:
1. Write and execute simple to complex analytical programs in R.
2. Demonstrate fundamental analytical packages in R.
3. Create tables and query from MongoDB.
4. Implement, configure and work with big data platform.
5. Install Hadoop and write Map Reduce Programs.
6. Apply data modeling using H2O packages.

REFERENCES:
OBJECTIVES:
- To get an overview of the full stack software and web development.
- To understand the object oriented structure and user interface programming through Python.
- To gain knowledge of web development using Flask Framework.
- To learn the web application deployment in real time scenarios.
- To learn to deploy the software in Linux and Windows platforms.

UNIT I  OBJECT ORIENTED APPROACH IN PYTHON

Suggested Activities:
- Flipped classroom on object oriented methods.
- Practical - Programming exercises involving the object oriented concepts.

Suggested Evaluation Methods:
- Quiz on object oriented methods
- Programming assignments.

UNIT II  USER INTERFACE APPLICATIONS IN PYTHON AND VERSION CONTROL SYSTEM

Suggested Activities:
- Flipped classroom on user interface programming models.
- Practical - Design of game with functional modules.
Suggested Evaluation Methods:
- Practical - Programming assignment on developing simple applications using wx Python.
- Quiz on windows elements and collaborative version control systems.
- Practical - Setting up a version control repository and the number of commits.

UNIT III FLASK FRAMEWORK FOR WEB DEVELOPMENT

Suggested Activities:
- Flipped classroom on HTML, shell commands and basic web development strategies
- Design of the Web layout
- Practical - Programming snippets and connection to the Mongodb database.

Suggested Evaluation Methods:
- Quiz on HTML basics, shell commands and running server with LAMP
- Programming assignment on Development of a web application with a connected database

UNIT IV REAL TIME DEPLOYMENT OF WEB APPLICATION
Deploy Web Applications with Flask and MongoDB – Example Applications – Blogs – Forums – Auto Evaluation of Student Assignments – Deployment Using AWS or Google Cloud or Heroku.

Suggested Activities:
- Flipped classroom on the development cycle of web.
- Programming and actual deployment of web applications.
- Use of git.

Suggested Evaluation Methods:
- Quiz on the cycle of web development.
- Porting the developed web applications in AWS/Google cloud/Heroku.
- Number of commits in git repository.

UNIT V DEPLOYMENT OF SOFTWARE IN LINUX AND WINDOWS PLATFORM

Suggested Activities:
- Flipped classroom on the method of pacakaging the software in Windows and Linux environments.
- Sample application deployment in Linux and Windows platform.

Suggested Evaluation Methods:
- Programming assignment on packaging the software developed from Unit I and Unit II.
- Deployment in Linux and Windows platform.
- Test cases.

TOTAL: 45 PERIODS
OUTCOMES:
On completion of the course, the student will be able to:
1. Understand the object oriented approach in Python.
2. Develop GUI applications with Python.
3. Use the collaborative version control system, git.
4. Package the developed code in Linux and Windows environment.
5. Deploy the developed web application using Flask in real time scenarios such as AWS.
6. Developer of the industrial software.

REFERENCES:
10. https://pypi.org/project/py2exe/0.9.2.0/

<table>
<thead>
<tr>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>CO2</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>CO3</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>CO4</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>CO5</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>CO6</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
</tbody>
</table>

CA5005 INTRODUCTION TO MACHINE LEARNING L T P C
3 0 0 3

OBJECTIVES:
- To understand the basic concepts of machine learning and probability theory.
- To appreciate supervised learning and their applications.
- To understand unsupervised learning like clustering and EM algorithms.
- To understand the theoretical and practical aspects of probabilistic graphical models.
- To learn other learning aspects such as reinforcement learning, representation learning, deep learning, neural networks and other technologies.
UNIT II    INTRODUCTION

Suggested Activities:
- Flipped classroom on Artificial Intelligence and Expert Systems.
- Practical - Installing Python and exploring the packages required for machine learning including numpy, scikit-learn, and matplotlib, IPython hmmppytk and pgmppy.

Suggested Evaluation Methods:
- Assignments on different types of learnings.
- Tutorials on probability theory.

UNIT II    SUPERVISED LEARNING

Suggested Activities:
- Flipped classroom on basics about classification and regression.
- Practical - Collection of data from different recourses and summarize the data.
- Practical - Build linear, multi-linear, logistic regression model to predict the data.

Suggested Evaluation Methods:
- Evaluation of the practical assignment against appropriate test sets.

UNIT III    UNSUPERVISED LEARNING

Suggested Activities:
- Flipped classroom on mixture models.
- External learning - Improving performance of the model using kernel methods.

Suggested Evaluation Methods:
- Assignments on mixture models.

UNIT IV    GRAPHICAL MODELS

Suggested Activities:
- Flipped classroom on Bayesian and Markov models.
- Practical - Implementation of Naive Bayes classifier for credit card analysis.
- Practical - Implement HMM for an application.
- External learning - Gaussian Processes and Topic Modeling.
Suggested Evaluation Methods:
- Quizzes on Markov model and HMM.
- Evaluation of the HMM application.

UNIT V       ADVANCED LEARNING

Suggested Activities:
- Flipped classroom on neural networks.
- Practical - Implement bagging approach for credit card analysis.
- External learning - Deep networks.

Suggested Evaluation Methods:
- Evaluation of the practical implementation.
- Assignments on deep networks.

TOTAL: 45 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
1. Choose and implement classification or regression algorithms for an application using an open source tool.
2. Implement probabilistic discriminative and generative algorithms for an application and analyze the results.
3. Use a tool to implement typical clustering algorithms for different types of applications.
4. Design and implement an HMM for a sequence model type of application.
5. Implement appropriate learning algorithms for any real time application using an open source tool.
6. Identify applications suitable for different types of machine learning with suitable justification.

REFERENCES:
OBJECTIVES:
- To learn the fundamentals of autonomous driving.
- To study the different ways of sensing internal states of Autonomous Ground Vehicles (AGVs).
- To learn the environment perception for autonomous driving.
- To explore the navigation techniques of AGVs.
- To learn the fundamentals of vehicle control systems and connected vehicles.

UNIT I  INTRODUCTION TO AUTONOMOUS DRIVING

Suggested Activities:
- Flipped classroom on autonomous driving system architecture.
- External learning - Building blocks of typical Unmanned Aerial Vehicles.
- Flipped classroom on robot operating system.
- External learning - Applications of autonomous vehicles (aerial, under water, ground vehicles).
- Assignment on the design requirement specifications of autonomous vehicles (aerial, under water, ground vehicles).

Suggested Evaluation Methods:
- Viva voce on assignment topics.
- Quizzes on Advanced Driver Assistance Systems (ADAS).
- Group Discussion on Google’s self-driving car.
UNIT II  SENSORS FOR AUTONOMOUS GROUND VEHICLES


Suggested Activities:
- Flipped Classroom on sensor characteristics.
- External learning - Working principle of IMU/GPS/RADAR sensors.
- External learning - Exploring Velodyne Lidar sensor dataset in Veloview software.

Suggested Evaluation Methods:
- Practical - Experiments on interfacing IMU sensor to Raspberry Pi board and recording the acceleration of a dummy vehicle.
- Practical - Experiments on interfacing Lidar/RADAR sensor to Raspberry Pi board and recording the distances to the nearby objects.
- Practical - Experiments on interfacing camera to Raspberry Pi board and capturing images/videos.

UNIT III  ENVIRONMENT PERCEPTION AND MODELING

Road Recognition: Basic Mean Shift Algorithm, Mean Shift Clustering, Mean Shift Segmentation, Mean Shift Tracking, Road Recognition Algorithm – Vehicle Detection and Tracking: Generating ROIs, Multi Resolution Vehicle Hypothesis, Vehicle Validation using Gabor Features and SVM, Boosted Gabor Features – Multiple Sensor Based Multiple Object Tracking.

Suggested Activities:
- Flipped classroom on basic mean shift algorithm.
- External learning - Lane detection algorithm.
- Flipped classroom on vehicle tracking.

Suggested Evaluation Methods:
- Practical - Implementation of Mean Shift Clustering / Mean Shift Segmentation Algorithm.
- Practical – Experiments on stationary obstacle detection algorithm using Lidar sensor.

UNIT IV  NAVIGATION FUNDAMENTALS


Suggested Activities:
- Flipped classroom on GPS orbits/GPS Signals.
- External learning - Indian Regional Navigation Satellite System (IRNSS).
- Assignment on the working principles of Google Map.

Suggested Evaluation Methods:
- Quizzes on GNSS signal structure.
- Viva Voce on assignment topics.
- Practical - Simulation of Waypoint Navigation Algorithm.
UNIT V VEHICLE CONTROL AND CONNECTED VEHICLE


Suggested Activities:
- Flipped classroom on cruise control.
- External learning - Study on proportional integral derivative (PID) control.
- Assignment - Communication protocols for connected vehicles.

Suggested Evaluation Methods:
- Viva Voce on assignment topic.
- Practical - Experiment on simple velocity control.
- Practical - Experiment on simple longitudinal motion control.

TOTAL: 45 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
1. Identify the requirements and design challenges of AGVs.
2. Select suitable sensors to sense the internal state and external world of AGVs.
3. Implement lane detection, road detection & vehicle detection algorithms.
4. Simulate/implement ground vehicle navigation algorithms.
5. Simulate/implement ground vehicle control systems.
6. Design communication protocols for connected vehicles.

REFERENCES:

<table>
<thead>
<tr>
<th></th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>CO2</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>CO3</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>CO4</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>CO5</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>CO6</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>3</td>
</tr>
</tbody>
</table>
OBJECTIVES:
- To learn the various E-learning approaches and components.
- To understand the types of design models of E-Learning.
- To explore the models for E-learning courseware development.
- To learn about E-learning authoring tools.
- To know about evaluation and management of E-learning solutions.

UNIT I INTRODUCTION

Suggested Activities:
- External learning - E-learning approaches and components.
- Discussion on blended learning.

Suggested Evaluation Methods:
- Assignment on E-learning approaches and components.
- Quizzes on blended learning.

UNIT II DESIGNING E-LEARNING COURSE CONTENT

Suggested Activities:
- Discussion forum on design models.
- External learning on E-Learning instructional methods.

Suggested Evaluation Methods:
- Assignment on design models of E-learning.
- Quiz on E-Learning instructional methods.

UNIT III CREATING INTERACTIVE CONTENT

Suggested Activities:
- Discussion on creation of story boards.
- Discussion on courseware creation.
- External learning - Types of authoring tools.

Suggested Evaluation Methods:
- Demonstration of Story Boards creation with Moodle.
- Demonstration of creation of a complete courseware with Moodle.
- Quiz on authoring tools.
UNIT IV  LEARNING PLATFORMS  

Suggested Activities:
- Discussion on LMS categories for E-learning.
- External learning - Functional areas of E-learning.

Suggested Evaluation Methods:
- Assignment on proprietary and open source LMS.
- Quiz on LMS solutions.

UNIT V  COURSE DELIVERY AND EVALUATION  

Suggested Activities:
- Discussion on planning and documentation.
- External learning - Evaluation and delivery methods.

Suggested Evaluation Methods:
- Assignment on planning and documentation.
- Quiz on evaluation and delivery methods.

OUTCOMES:
On completion of the course, the students will be able to:
1. Distinguish the phases of activities in the models of E-learning.
2. Identify appropriate instructional methods and delivery strategies.
3. Choose appropriate E-learning authoring tools.
5. Evaluate the E-learning courseware.
6. Manage the E-learning courseware.

REFERENCES:
OBJECTIVES:
- To introduce the basics and necessity of software testing.
- To provide various testing techniques along with concepts of software bugs and its impact.
- To develop and validate a test plan.
- To build a testing team required.
- To understand the need for and challenges in test automation and to develop testing scripts.

UNIT I  TESTING PRINCIPLES AND AXIOMS  9

Suggested Activities:
- Flipped classroom on testing axioms.
- Identify and analyze syntax error, semantic error, bug and defect for programs.

Suggested Evaluation Methods:
- Quiz and discussion on testing axioms.
- Identifying fallacies in requirements specification.
- Identify the various types of errors, bugs and defects for a case study.

UNIT II  BLACK BOX, WHITE BOX TESTING AND TEST ADEQUACY  9
Suggested Activities:
- Flipped classroom on test adequacy criteria.
- External learning - Exploring white box testing tools like veracode, eclemma, rcunit, cppunit, Junit, JUnit etc.
- Analyzing the cyclomatic complexity of code segments.

Suggested Evaluation Methods:
- Quiz and discussion on cyclomatic complexity.
- Assignments on white box testing tools like Selenium, Appium, Robotium and carrying out simple BBT and WBT using tools.
- Solving problems related to cyclomatic complexity.

UNIT III LEVELS OF TESTING


Suggested Activities:
- External learning - Exploring the integration testing tools for various programming languages – VectorCAST/C++, CITRUS (Java), FitNesse (open source), Rational test integration tester, Protractor (Angular, Angular JS), Jasmine (JavaScript), Spock (Java) and the regression testing tools – Sahi Pro, Watir, IBM Rational Regression Tester, TestDrive etc.
- Flipped classroom on alpha and beta testing.
- Analyzing various levels of testing required for a software product.

Suggested Evaluation Methods:
- Assignments on integration testing tools and regression testing tools.
- Quiz and discussion on alpha and beta testing.
- Identifying and performing various levels of testing for a case study.

UNIT IV TEST MANAGEMENT


Suggested Activities:
- Flipped classroom on reporting test results.
- External learning - Exploring the organization structures and organizational behaviour in the context of software testing.
- Analyzing how to build testing groups for various types of projects and organizations.

Suggested Evaluation Methods:
- Quiz and discussion on reporting test results.
- Finding out the organization structure and organizational behaviour for given case studies.
- Building test groups for given case studies.
UNIT V TEST AUTOMATION


Suggested Activities:
- Flipped classroom on test metrics and measurements.
- External learning - Exploring the risks involved in automated testing and exploring the ways to improve your testing skills apart from using testing tools.
- Practical - Install and learn popular software testing tools like Selenium, Win Runner, LoadRunner, Performance Tester etc.
- Learning to write test scripts.

Suggested Evaluation Methods:
- Quiz and discussion on test metrics and measurements.
- Assignments on evaluating the risks involved in automated testing for given case studies.
- Assignments on writing test scripts to carry out various types of testing in test automation tools.

TOTAL: 45 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
1. Obtain an insight to software testing.
2. Apply both black box testing and white box testing.
3. Understand and apply multiple levels of testing.
4. Understand the role of a tester as an individual and as a team member.
5. Apply software testing for large projects using automated testing tools.
6. Maintain documentation on testing.

REFERENCES:
OBJECTIVES:
- To understand the basic ideas and principles of neural networks.
- To understand the basic concepts of deep learning.
- To familiarize with image processing facilities like TensorFlow and Keras.
- To appreciate the use of deep learning applications.
- To understand and implement deep learning architectures.

UNIT I  BASICS OF NEURAL NETWORKS  9
Basic Concept of Neurons – Perceptron Algorithm – Feed Forward and Backpropagation Networks.

Suggested Activities:
- Discussion of role of neural networks.
- External learning - Boltzmann Machine, perceptron.
- Practical - Implementation of simple neural network in Matlab

SUGGESTED EVALUATION METHODS
- Tutorials on perceptron.
- Assignments on backpropagation networks.
- Quizzes on neural networks.

UNIT II  INTRODUCTION TO DEEP LEARNING  9

Suggested Activities:
- Discussion of role of Gradient Descent in deep learning.
- External learning - Feature extraction and feature learning.
- Practical - Implementation of TensorFlow and Keras applications.
Suggested Evaluation Methods:
- Tutorials on gradient descent and regularization
- Assignments on optimization.
- Quizzes on deep learning regularization and optimization.

UNIT III  CONVOLUTIONAL NEURAL NETWORKS
9

Suggested Activities:
- Discussion of role of convolutional networks in Machine Learning.
- External learning - Concept of convolution and need for Pooling.

Suggested Evaluation Methods:
- Tutorials on image classification and recurrent nets.
- Assignments on image classification performances.
- Quizzes on convolutional neural networks.

UNIT IV ADDITIONAL DEEP LEARNING ARCHITECTURES 9

Suggested Activities:
- Discussion of role of Deep Learning architectures.
- External learning - Compression of features using Auto-encoders.
- Practical - Implementation of simple deep learning architectures.

Suggested Evaluation Methods:
- Tutorials on LSTM and Autoencoders.
- Assignments on deep generative models, Deep Belief Networks.
- Quizzes on deep learning architectures.

UNIT V APPLICATIONS OF DEEP LEARNING 9

Suggested Activities:
- Discussion of role of deep learning in image and NLP applications.
- External learning - NLP concepts.
- Practical - Implementation of simple deep learning for object detection and recognition in images.

Suggested Evaluation Methods:
- Tutorials on images segmentation.
- Assignments on parsing and sentiment analysis.
- Quizzes on deep learning applications
OUTCOMES:
On completion of the course, the students will be able to:
- Understand the role of deep learning in machine learning applications.
- Get familiar with the use of TensorFlow and Keras in deep learning applications.
- Design and implement deep learning applications.
- Critically analyze different deep learning models in image related projects.
- Know about applications of deep learning in NLP and image processing.

REFERENCES:

<table>
<thead>
<tr>
<th></th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>CO2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>CO3</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>CO4</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>CO5</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>CO6</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>

CA5010 GAME PROGRAMMING TECHNIQUES L T P C
3 0 0 3

OBJECTIVES:
- To know the basics of 2D and 3D graphics for game development.
- To know the stages of game development.
- To understand the basics of game engine.
- To survey the gaming development environment and toolkits.
- To learn and develop simple games using Pygame environment.

UNIT I 3D GRAPHICS FOR GAME PROGRAMMING
Suggested Activities:
- Discussion about computer and video games origin and history.
- Discussion of graphics objects, Open source language for Game development like Pygame and Processing.py - a Language for Creative Arts.
- External learning - Practical problems in translation, scaling, zooming and rotation of 2D and 3D objects.
- Practical - Installation of Pygame and Pygame Zero and Implementation of colour models and shading models in Python.

Suggested Evaluation Methods:
- Tutorial - 2D and 3D transformations.
- Evaluation of programming exercises for Python implementation.
- Assignments on image projections and colour models.
- Quizzes on 2D and 3D game object transforms.

UNIT II GAME DESIGN PRINCIPLES

Suggested Activities:
- Flipped classroom on animation.
- Creation of game script in natural language and story creation.
- External learning - Practical problems in game level design.
- Practical - Producing game level design document, detailed document.

Suggested Evaluation Methods:
- Tutorial - Script writing.
- Assignments on game proposal writing.
- Quizzes on game design document.

UNIT III GAME ENGINE DESIGN

Suggested Activities:
- Flipped classroom on rendering.
- External learning - Image rendering and animation.
- Practical - Implementation of simple animations in Pygame and Processing.py

Suggested Evaluation Methods:
- Tutorial problems in collision detection.
- Assignments on game AI and path finding.
- Quizzes on rendering.

UNIT IV OVERVIEW OF GAMING PLATFORMS AND FRAMEWORKS
Pygame Game development – Unity – Unity Scripts – Mobile Gaming, Game Studio, Unity – Single player and Multi-Player games.
Suggested Activities:
- Flipped classroom on gaming environments.
- External learning on Unity Game Engine.
- Practical - Installation of Unity and scripts.
- Practical - Pygame routines for character rendering, transformations and sound processing.

Suggested Evaluation Methods:
- Tutorial - Collision detection.
- Assignments on Unity Game Engine.
- Quizzes of all topics related to Unity and Pygame.

UNIT V   GAME DEVELOPMENT USING PYGAME


Suggested Activities:
- External learning - Writing Unity scripts and assets.
- Practical - Implementation of simple games.

Suggested Evaluation Methods:
- Tutorial problems in 2D and 3D graphics programming.
- Programming problems like asset creation
- Quizzes on game development in Pygame.

TOTAL: 45 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
1. Have knowledge on the concepts of 2D and 3D graphics.
2. Know about games and their genres with their origin and history.
3. Prepare game design documents.
4. Understand the implementation of gaming engines.
5. Survey gaming environments and frameworks.
6. Implement a simple game in Pygame.

REFERENCES:
OBJECTIVES:
- To enrich student learning in multimedia systems.
- To train the students to acquire knowledge in multimedia related technologies.
- To acquire knowledge about multimedia techniques to enhance quality of service.
- To acquire knowledge in the development of multimedia systems.
- To learn about the multimedia elements in a comprehensive way.

UNIT I
INTRODUCTION TO MULTIMEDIA ELEMENTS

Suggested Activities:
- Flipped classroom on multimedia concepts.
- Practical - Creating and editing visual elements using tools like Audacity, Fontographer, Blender, Photoshop and flash.

Suggested Evaluation Methods:
- Demonstration of the practical exercise.
- Assignments on creativity and visual appearance.
- Quizzes on sound, speech and image-related concepts.

UNIT II
MULTIMEDIA COMPRESSION

Suggested Activities:
- Flipped classroom on different compression techniques.
Suggested Evaluation Methods:
- Demonstration, finalization and output of the practical learning.
- Quizzes on MPEG and audio encoding.

UNIT III MULTIMEDIA ARCHITECTURES 9

Suggested Activities:
- Flipped classroom on concepts of Multimedia hardware architectures.
- External learning - Digital repositories.

Suggested Evaluation Methods:
- Tutorial - Document architecture.
- Quizzes on hypermedia.

UNIT IV MULTIMEDIA OPERATING SYSTEM AND DATABASES 9

Suggested Activities:
- Flipped classroom on multimedia database and indexing structures.
- External learning - Data structures for storing multimedia data.

Suggested Evaluation Methods:
- Quizzes on various concepts of multimedia databases.
- Assignments on various operations on data

UNIT V MULTIMEDIA COMMUNICATION & APPLICATIONS 9

Suggested Activities:
- Practical - Designing user interfaces and developing simple games.
- External learning - Mixed reality.

Suggested Evaluation Methods:
- Demonstration of developed applications.
- Quizzes on virtual reality and augmented reality.

TOTAL: 45 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
1. Handle the multimedia elements effectively.
2. Encode and decode the multimedia elements.
3. Understand the underlying multimedia computing architectures used for media development.
4. Develop effective strategies to deliver quality-of-experience in multimedia applications.
5. Design and implement algorithms and techniques related to multimedia objects.
6. Design and develop multimedia applications in various domains.
REFERENCES:

<table>
<thead>
<tr>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>CO2</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>CO3</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>CO4</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>CO5</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>CO6</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>2</td>
</tr>
</tbody>
</table>

CA5012 DATA VISUALIZATION TECHNIQUES L T P C 3 0 0 3

OBJECTIVES:
- To understand the fundamentals of data visualization.
- To know the working principles of various information visualization tools.
- To acquire knowledge about the issues in data representation.
- To visualize the complex engineering design.
- To gain skill in designing real time interactive information visualization system.

UNIT I INTRODUCTION 9

Suggested Activities:
- Blended Learning - Displaying different types visualization images.
- Flipped classroom on the task of representing information.
- External learning - Practical problems related to acquiring data.
- Practical - Representing various varieties of data.

Suggested Evaluation Methods:
- Tutorial - Different data visualizing images.
- Assignment on different data acquiring methods.
- Quizzes on various issues and solutions in different visualization applications.
- Demonstration of the techniques used for data representation.
UNIT II DATA REPRESENTATION


Suggested Activities:
- Blended Learning - Human visual and auditory system.
- Flipped classroom on color formats.
- Practical - Implementation of the interactive forms.
- External learning - Survey on different human computer interaction and types of user interface.

Suggested Evaluation Methods:
- Assignments on human visual and auditory system.
- Quizzes on color format.
- Assessments design and creativity.
- Assignments on various human computer interaction user interface.

UNIT III DATA PRESENTATION


Suggested Activities:
- Blended Learning - Drawing charts for display.
- Flipped classroom on various presentation techniques.
- External learning - Different font and font styles, symbols and gesture representation.
- Practical - Implementation of these presentations through interfaces in computers.

Suggested Evaluation Methods:
- Assignment on chart preparation.
- Tutorial - Various presentation techniques.
- Assignment on gesture presentation.
- Demonstration of the designed interface layout.

UNIT IV INTERACTION


Suggested Activities:
- Flipped classroom on various interacting Techniques.
- Practical - Implementations of interactive interfaces.
- External learning - Interaction facilities and high level support for animation design.

Suggested Evaluation Methods:
- Tutorial - Interaction models.
- Demonstration of the based on interactivity.
- Assignment on animation design.

UNIT V CURRENT TRENDS

Suggested Activities:

- Practical - Mini project for designing and implementing innovative interfaces.
- Flipped classroom on the implementation of virtual reality environment.

Suggested Evaluation Methods:

- Demonstration of the mini project.
- Tutorial - Virtual reality application.

TOTAL: 45 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:

1. Apply mathematics and basic science knowledge for designing information visualizing system.
2. Collect data ethically and solve engineering problem in visualizing the information.
3. Implement algorithms and techniques for interactive information visualization.
4. Conduct experiments by applying various modern visualization tool and solve the space layout problem.
5. Analyze and design system to visualize multidisciplinary multivariate Data individually or in teams.
6. Develop a cost effective and a scalable information visualization system.

REFERENCES:


<table>
<thead>
<tr>
<th>CO1</th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>CO2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>CO3</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>CO4</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>CO5</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>CO6</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>
OBJECTIVES:

- To learn about the design of the Unix operating system.
- To become familiar with the various data structures used in UNIX.
- To understand the Unix file system and its system calls.
- To study about process management and scheduling in operating system.
- To learn about memory management and I/O systems.

UNIT I  OVERVIEW


Suggested Activities:

- Flipped classroom on operating system services.
- Practical:
  - Implement the system call ‘cat’ using command line arguments and generate the executable version of the program and invoke the executable file using exec system calls (fork, wait etc).
  - Implement a scenario resulting to an incorrect linked list because of context switch.
  - Implement the five scenarios in the getblk algorithm by using first in first out scheme.
  - Simulate the function of bread(), breada(), bwrite and brelse.

Suggested Evaluation Methods:

- Quiz on operating system services.
- Evaluation of the functions implemented.

UNIT II  FILE SUBSYSTEM

Internal Representation of Files: inodes – Structure of a Regular File – Directories – Conversion of a Path Name to an Inode – Super Block – Inode Assignment to a New File – Allocation of Disk Blocks.

Suggested Activities:

- Flipped classroom on files and directory structure.
- Practical:
  - Implement the five scenarios in the iget algorithm by using least recently used scheme.
  - Implement the bmap algorithm and find the block number and the byte offset in file system for the given offset. Assume the disk block contain 1024 bytes.
    - 96000
    - 9999999
  - Simulate the function of iput, ialloc, ifree, alloc and ifree.
  - Write a program to display the directory entries (i.e., byte offset, inode number and the file name).

Suggested Evaluation Methods:
Quiz on files and directory structure.
Evaluation of the functions implemented.

UNIT III SYSTEM CALLS FOR THE FILE SYSTEM

Suggested Activities:
- Flipped classroom on file system and system calls.
- Practical -
  - How does the command mkdir work? (Hint: When mkdir completes, what are the inode numbers for “.” and “..”?).
  - Simulate the function of chown, chmod, stat and fstat.
  - Set the whole-file lock with fcntl() and lockf().
  - Write a program to print the mount table whenever an external device is connected to the Unix system.

Suggested Evaluation Methods:
- Quiz on file system calls.
- Evaluation of the functions implemented.

UNIT IV PROCESSES

Suggested Activities:
- Flipped classroom on context switching
- Practical -
  - Implement the algorithm for allocating and freeing memory pages and page tables. Which data structures would allow best performance?
  - Design an algorithm that translates virtual address to physical addresses, given the virtual address and the address of the region entry.
  - Implement an algorithm that exchange messages over pipe (use of pipe and dup and fork).
  - Write a program to communicate between two process using signals.

Suggested Evaluation Methods:
- Quiz on Context switching.
- Evaluation of the functions implemented.

UNIT V MEMORY MANAGEMENT AND I/O

Suggested Activities:
- Flipped classroom on virtual memory concepts
- Practical -
  - Write a program that tracks the allocation of space on a swap device.
Write a program that verifies that the file systems on a disk do not overlap. The program should take two arguments: a device file that represents a disk volume and a descriptor file that gives section numbers and section lengths for the disk type. The program should read the super blocks to make sure that file systems do not overlap.

- Implement sty command: with no parameters, it retrieves the values of terminal settings and report them to the user.
- Encode a line disciple that writes the machine name at the beginning of each line of output.

**Suggested Evaluation Methods:**
- Quiz on virtual memory concepts.
- Evaluation of the functions implemented.

**TOTAL: 45 PERIODS**

**OUTCOMES:**
On completion of the course, the students will be able to:
1. Understand UNIX architecture and describe the component of operating system.
2. Explain how they interact with computer hardware.
3. Gain a deeper understanding of system calls in Unix operating system.
4. Apply the concepts of operating systems design to practical problems.
5. Design and implement the subsystems of an operating system.
6. Critically analyze different data structures and algorithms used in the building of a kernel.

**REFERENCES:**
OBJECTIVES:
- To learn the technologies of the .NET framework.
- To cover all segments of programming in C# starting from the language basis, followed by the object oriented programming concepts.
- To update and enhance skills in writing Windows applications, ADO.NET and ASP.NET.
- To introduce advanced topics namely data connectivity, WPF, WCF and WPF with C# and .NET 4.5.
- To implement mobile applications using .Net Compact Framework.

UNIT I C# LANGUAGE BASICS

Suggested Activities:
- Installation of .Net framework and experimenting simple C# programs using IDE.
- Flipped Classroom on CLR internals.
- Creation of shared assemblies.

Suggested Evaluation Methods:
- Quiz on CLR internals.
- Tutorials on C# programming fundamentals.

UNIT II C# ADVANCED FEATURES

Suggested Activities:
- Implementing delegates and handling events.
- Practical - Generic collections, memory management and exception handling.

Suggested Evaluation Methods:
- Demonstration of implemented programs.
- Tutorial case studies on advanced C# features.

UNIT III BASE CLASS LIBRARIES AND DATA MANIPULATION

Suggested Activities:
- Implementation of Threads and Synchronization based application.
- Practical - Programs on XML and operations using parsers.
- Application development with ADO.NET.
Suggested Evaluation Methods:
- Tutorials on SAX and DOM parsers.
- Presentation of ADO.NET based application.

UNIT IV  WINDOW AND WEB BASED APPLICATIONS  9

Suggested Activities:
- Practical - Programs using ASP.NET and State management controls.
- Flipped classroom on web services with .NET.
- Tutorials on WCF framework.

Suggested Evaluation Methods:
- Quizzes.
- Demonstration of the implemented programs on ASP.NET web services.

UNIT V  .NET COMPACT FRAMEWORK  9

Suggested Activities:

Suggested Evaluation Methods:
- Presentation of .NET compact framework application.

OUTCOMES:
On completion of the course, the student will be able to:
1. Understand the difference between .NET and Java framework.
2. Work with the basic and advanced features of C# language.
3. Create applications using various data providers.
4. Create web application using ASP.NET.
5. Create mobile application using .NET compact framework.
6. Integrate all the features of C# language and build complex web applications in .NET framework.

TOTAL: 45 PERIODS

REFERENCES:
OBJECTIVES:
- To analyze various software architectures and understand the basic principles of service orientation.
- To learn the service oriented architecture and micro services architecture.
- To understand the technologies associated with SOA.
- To analyze and implement web service based applications and realize SOA.
- To learn micro services related frameworks and develop applications.

UNIT I  SOFTWARE ENGINEERING PRACTICES 9

Suggested Activities:
- Sample application for each type of architecture.
- Study of popular enterprise applications.
- Cloud computing platforms comparison.
- DevOps solution fundamentals.

Suggested Evaluation Methods:
- Quiz on various concepts.
- Simple development based on the solutions and study.

UNIT II  SOA AND MSA BASICS 9

Suggested Activities:
- Applications of SOA and MSA.
- OOAD and SOAD comparison.
- Identifying simple services based on SOA and MSA.
Suggested Evaluation Methods:
- Case studies of various SOA applications.
- Application based comparison.

UNIT III WEB SERVICES
9

Suggested Activities:
- XML processing.
- Exploring the structure of SOAP, WSDL and UDDI.
- Creation of web services in Java/.NET/Python environment.
- RESTful web services.
- Study of middleware services for IoT.

Suggested Evaluation Methods:
- Implementing XML, DOM and SAX.
- Programming exercises.

UNIT IV SERVICE ORIENTED ANALYSIS AND DESIGN
9

Suggested Activities:
- Study of business process services.
- Orchestration of Web services.

Suggested Evaluation Methods:
- Quiz on service design principles.
- Demonstration - Orchestrated web services.

UNIT V MICROSERVICE BASED APPLICATIONS
9

Suggested Activities:
- Implementation of microservices architecture with Python.
- Creation of container services.
- Cloud deployment.

Suggested Evaluation Methods:
- Micro service based application case study.
- Cloud deployment in different platforms.

TOTAL: 45 PERIODS
OUTCOMES:
On completion of the course, the students will be able to:
1. Understand different types of software architecture.
2. Understand the need for MSA over SOA.
3. Understand the XML based standards associated with SOA.
4. Analyze and design SOA based applications.
5. Create Microservices using different software frameworks.
6. Integrate various microservices for realizing enterprise like application.

REFERENCES:

<table>
<thead>
<tr>
<th>CO1</th>
<th>CO2</th>
<th>CO3</th>
<th>CO4</th>
<th>CO5</th>
<th>CO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>PO1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>PO2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>PO3</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>PO4</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>PO5</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>PO6</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>

OBJECTIVES:
- To develop an awareness of the need for project planning and management.
- To know about software effort estimation and activity planning.
- To explore risk and people management.
- To learn about project monitoring and control mechanisms.
- To know about software quality management.
UNIT I  INTRODUCTION

Suggested Activities:
- Discussion on software project management planning.
- External learning - Process models.

Suggested Evaluation Methods:
- Assignment on project management framework.
- Quiz on process models.

UNIT II  SOFTWARE EFFORT ESTIMATION AND ACTIVITY PLANNING

Suggested Activities:
- Discussion on software effort estimation methods.
- External learning - Software activity planning.

Suggested Evaluation Methods:
- Quiz on software effort estimation methods.
- Assignment on activity planning of a case study.

UNIT III  SOFTWARE RISK AND PEOPLE MANAGEMENT

Suggested Activities:
- Discussion on risk management approaches.
- External learning on People Management.

Suggested Evaluation Methods:
- Assignment on risk management.
- Quiz on people management.

UNIT IV  SOFTWARE PROJECT MONITORING AND CONTROL
Suggested Activities:
- Discussion on project monitoring.
- External learning - Software control mechanisms.

Suggested Evaluation Methods:
- Assignment on project monitoring.
- Quiz on software control mechanisms.

UNIT V SOFTWARE QUALITY MANAGEMENT

Suggested Activities:
- Discussion on components of Software Quality Management.
- External learning on Software Quality measures.

Suggested Evaluation Methods:
- Assignment on various SQM standards and bodies.
- Quiz on software quality measures.

TOTAL: 45 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
1. Differentiate between various software process models.
2. Prepare project planning documents.
3. Estimate the software cost for projects.
4. Perform effective activity planning.
5. Prepare effective project scheduling work product.
6. Perform software quality management activities.

REFERENCES:
OBJECTIVES:
- To impart the fundamental aspects and principles of mixed reality technologies.
- To know the internals of the hardware and software components involved in the development of mixed reality enabled applications.
- To learn about the graphical processing units and their architectures.
- To gain knowledge about mixed reality application development.
- To evaluate the mixed reality based applications.

UNIT I INTRODUCTION

Suggested Activities:
- Flipped classroom on the use of MR applications.
- Experience the virtual reality effect by watching videos.
- Assignment on comparison of VR with traditional multimedia applications.

Suggested Evaluation Methods:
- Tutorial - MR applications.
- Brainstorming session - VR effects.
- Quizzes on difference between VR and Multimedia applications.
UNIT II  MR COMPUTING ARCHITECTURE

Suggested Activities:
- Flipped classroom on basic graphics pipeline.
- External learning - Different types of graphics architectures and workstations.
- Practical - GPU programming.

Suggested Evaluation Methods:
- Tutorial - Graphics pipeline.
- Brainstorming session - Graphics architectures.
- Quizzes on various topics of the unit.
- Demonstration of GPU programs for creating simple multimedia Applications.

UNIT III  MR MODELING

Suggested Activities:
- Flipped classroom on modeling three dimensional objects.
- External learning - Collision detection algorithms.
- Practical - Creating three dimensional models.

Suggested Evaluation Methods:
- Tutorial - 3D modeling techniques.
- Brainstorming session - Collision detection algorithms.
- Demonstration of three dimensional models.

UNIT IV  MR PROGRAMMING

Suggested Activities:
- External learning - Different types of programming toolkits.
- Practical - Create VR scenes using toolkits like World ToolKit, Java 3D, Ghost, People Shop, Unity.

Suggested Evaluation Methods:
- Tutorial sessions on different programming toolkits for MR.
- Demonstration of MR scene creation.
UNIT V  APPLICATIONS


Suggested Activities:
- External learning - Learn different types of available MR applications.
- Practical - Develop MR application in any domain of your interest.
- Tutorial - MR applications

Suggested Evaluation Methods:
- Evaluation of the developed MR application.
- Demonstration of MR application development and appropriate evaluation.

TOTAL: 45 PERIODS

OUTCOMES:
On completion of the course, the student will be able to:
1. Discuss the basic concepts of Mixed Reality.
2. Design and develop the Mixed Reality applications in different domains.
3. Design various models using modeling techniques.
5. Understand the working principles of input output devices used in mixed reality applications.
6. Evaluate mixed reality based applications.

REFERENCES:

<table>
<thead>
<tr>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>CO2</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>CO3</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>CO4</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>CO5</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>CO6</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>2</td>
</tr>
</tbody>
</table>
OBJECTIVES:
- To learn the basic concepts of digital image processing and various image transforms.
- To familiarize the student with the image enhancement techniques.
- To expose the student to a broad range of image processing techniques and their applications.
- To appreciate the use of current technologies that are specific to image processing systems.
- To expose the students to real-world applications of image processing.

UNIT I  FUNDAMENTALS OF IMAGE PROCESSING

Suggested Activities:
- Discussion on image processing applications.
- External learning - Open source tools like Octave/SciLab/OpenCV , types of images.
- Practical - Reading and writing of images in Matlab and OpenCV/Octave/SciLab.

Suggested Evaluation Methods:
- Tutorials on image operations, image connectivity and distance measures.
- Assignments on sampling, quantization and image operations.
- Quizzes on image types.

UNIT II  IMAGE ENHANCEMENT

Suggested Activities:
- Discussion of mathematical transforms.
- Numerical problem solving using Fourier transform.
- External learning - image noise and types of noises.
- Practical - Implementation of simple spatial filters like low pass filters and high pass filters in Matlab/OpenCV.

Suggested Evaluation Methods:
- Tutorials on image transforms, image smoothing.
- Assignments on histogram specification and equalization, spatial filters.
- Quizzes on noise modeling.

UNIT III  IMAGE RESTORATION AND MULTI-RESOLUTION ANALYSIS
Suggested Activities:
- Discussion on image artifacts and blur.
- Discussion on the role of wavelet transforms in filter and analysis.
- Practical - Implementation of noise modeling in Matlab/Octave/SciLab.
- Practical - Implementation of wavelet transforms and deconvolution algorithms in Matlab/Octave.

Suggested Evaluation Methods:
- Tutorials on wavelet transforms.
- Assignments on order statistics filters and multi resolution expansions.
- Quizzes on wavelet transforms.

UNIT IV IMAGE SEGMENTATION AND FEATURE EXTRACTION


Suggested Activities:
- Flipped classroom on importance of segmentation.
- External learning - Discussion of features, feature selection and reduction.
- Practical - Implementation of SIFT, SURF in Matlab/Octave/SciLab.
- Practical - Implementation of PCA in Matlab/Octave.

Suggested Evaluation Methods:
- Tutorials on image segmentation and edge detection.
- Assignments on feature extraction and reduction.
- Quizzes on feature selection and extraction.

UNIT V IMAGE PROCESSING APPLICATIONS


Suggested Activities:
- Discussion on machine learning in image processing.
- Discussion on image classifiers.
- Discussion on biometrics such as iris, fingerprint and face recognition.
- Discussion on image security such as steganography and digital watermarking.
- External learning - Medical imaging and remote sensing.
- External learning - Study of visual effects and Forensic applications.
- Practical - Image classifier using SVM in Matlab/Octave.
- Practical - Extraction of features in fingerprint using Matlab/Octave.

Suggested Evaluation Methods:
- Tutorials on image classifier and clustering.
- Assignments on support vector machines and EM algorithm.
- Quizzes on image processing applications.

TOTAL: 45 PERIODS
OUTCOMES:
On completion of the course, the students will be able to:
1. Implement basic image processing operations.
2. Apply and develop new techniques in the areas of image enhancement and restoration.
3. Understand the image segmentation algorithms.
4. Extract features from images.
5. Apply classifiers and clustering algorithms for image classification and clustering.
6. Design and develop an image processing application that uses different concepts of image processing.

REFERENCES:

<table>
<thead>
<tr>
<th></th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>CO2</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>CO3</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>CO4</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>CO5</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>CO6</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>

CA5019 TEXT MINING TECHNIQUES

OBJECTIVES:
- To understand the basic issues and needs of text mining.
- To appreciate the different aspects of text categorization and clustering.
- To understand the role played by text mining in information retrieval and extraction.
- To appreciate the use of probabilistic models and its principles applicable in text mining.
- To appreciate the current trends in text mining on various systems.

UNIT I INTRODUCTION
Suggested Activities:
- Develop a web application for search engine.
- Tokenize the given text information using any parser.
- Practical - Implement all the preprocessing steps needed for text mining.

Suggested Evaluation Methods:
- Evaluation of the implementations the preprocessing steps in laboratory environment.

UNIT II TEXT CATEGORIZATION AND CLUSTERING 10

Suggested Activities:
- Role playing to be carrying out for grouping the students to understand the working principles of clustering and classification.

Suggested Evaluation Methods:
- Assignments on analyzing the performance of different clustering and classification algorithms and show the best performance of each algorithm for any specific application.

UNIT III TEXT MINING FOR INFORMATION RETRIEVAL AND INFORMATION EXTRACTION 10

Suggested Activities:
- In-class activity - Name Entity and relation extraction using role play game.
- In-class activity - Show the working principle of searching technique.

Suggested Evaluation Methods:
- Assignments on developing flash or animated presentation for explaining the working principles of any one algorithm for information retrieval and extraction.

UNIT IV PROBABILISTIC MODELS 9

Suggested Activities:
- In-class activity - Document clustering and information extraction.
- External learning - Markov models and entropy models.
Suggested Evaluation Methods:
- Tutorial - Topic modeling to show its behavior on different data types.

UNIT V RECENT TRENDS

Suggested Activities:
- In-class activity - Visualization approaches.
- External learning - Text mining applications and case studies.

Suggested Evaluation Methods:
- Assignments on extracting the sentiment expressed in any given sentence using opinion word.
- Tutorial - Methodologies available to detect the spam in opinion mining.

TOTAL: 45 PERIODS

OUTCOMES:
On completion of the course, the student will be able to:
1. Identify the different features that can be mined from text and web documents.
2. Use available open source classification and clustering tools on some standard text data sets.
3. Modify existing classification or clustering algorithms in terms of functionality or features used.
4. Design a system that uses text mining to improve the functions of an existing open source search engine.
5. Implement a text mining system that can be used for an application of your choice.
6. Use the opinion mining concepts to extract the sentiment from the large database.

REFERENCES:
OBJECTIVES:
- To get exposed to the concepts of data warehousing architecture and implementation.
- To conceptualize data mining and the need for pre-processing and to analyze the mining techniques for realistic data.
- To characterize the kinds of patterns that can be discovered by association rule mining.
- To implement classification and clustering techniques on large datasets.
- To identify business applications and trends of data mining.

UNIT I DATA WAREHOUSE

Suggested Activities:
- Assignments on data warehouse modeling using a real time scenario.
- Assignment on describing the similarities and the differences of the multidimensional models and analyzing their advantages and disadvantages with regard to one another.
- Practical - Implementing various OLAP operations on a multidimensional data.
- Practical - Execute multidimensional data model using SQL queries.
- Discussion on the advantages of indexing structures.

Suggested Evaluation Methods:
- Tutorial - Case study on OLAP schema level representation and OLAP operations.
- Assignment on OLAP operations and schema level representation.
- Tutorial - Building a data warehouse using open source tools such as Talend.

UNIT II DATA MINING & DATA PREPROCESSING

Suggested Activities:
- Discussion on knowledge discovery database.
- Assignments on numerical problems on smoothing, normalization and attribute subset selection.
- Evaluate attribute relevance analysis on a real time application data warehouse.
- Evaluate information gain of an attribute in a real time database.

Suggested Evaluation Methods:
- Tutorial - Data cleaning and data transformation.
- Assignments on data integration and transformation.
- Assignment on data reduction and data discretization.
- Quizzes on data preprocessing.
UNIT III ASSOCIATION RULE MINING

Introduction – Data Mining Functionalities – Association Rule Mining – Mining Frequent Itemsets with and without Candidate Generation – Mining Various Kinds of Association Rules – Constraint – Based Association Mining.

Suggested Activities:
- Discussion and problem solving of different association rule mining algorithms (Apriori algorithms and FP-Growth algorithms).
- Practical - Implementation of association rule mining using Data mining tools such as Weka.
- Practical - Comparing the performance of each algorithm with various kinds of large data sets.

Suggested Evaluation Methods:
- Quizzes on different data mining functionalities and types of association rule mining.
- Tutorial - Different real time applications of association rule mining.

UNIT IV CLASSIFICATION & PREDICTION

Classification versus Prediction – Data Preparation for Classification and Prediction – Classification by Decision Tree – Bayesian Classification – Rule Based Classification – Classification by Back Propagation – Support Vector Machines – Associative Classification – Lazy Learners – Other Classification Methods – Prediction – Accuracy and Error Measures – Evaluating the Accuracy of a Classifier or Predictor – Ensemble Methods – Model Section.

Suggested Activities:
- Discussion on tree pruning.
- Assignments on calculation of the computational complexities and accuracy of the classification algorithms.
- Discussion on different real-time applications of classification and evaluating the accuracy of a classifier.
- Assignments on problem solving of classification algorithms.
- Comparative study on different classification algorithms.

Suggested Evaluation Methods:
- Quizzes on different classification methods.
- Tutorial - Accuracy and error measures different classification methods.
- Assignment on support vector machines.

UNIT V CLUSTERING


Suggested Activities:
- Comparative study on the various clustering algorithms.
- Discussion on real time applications of outlier analysis.
- Practical - Implementation of clustering algorithms using data mining tools.
- Practical - Design and implementation of a clustering method that finds clusters in large data cubes effectively and efficiently.
- Assignments on comparative study of clustering algorithms in terms of the following criteria: shapes of clusters that can be determined by input parameters that must be specified and limitations.
Assignments on categorization such as to categorize the kinds of constraints that can be imposed on the clusters produced and discuss how to perform clustering efficiently under such kinds of constraints.

Practical - Develop an application where the border between normal objects and outliers is often unclear, so that the degree to which an object is an outlier has to be well estimated.

Suggested Evaluation Methods:
- Quizzes different types of clustering methods.
- Tutorial - High-dimensional data clustering.
- Assignment on density based, grid based and model based clustering methods.

TOTAL: 45 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
1. Design, create and maintain data warehouses.
2. Apply data mining techniques and methods to large data sets.
3. Evaluate various mining techniques on complex data objects.
4. Evolve multidimensional intelligent model from typical system.
5. Discover the knowledge imbibed in the high dimensional system.
6. Understand various tools of data mining and their techniques to solve the real time problems.

REFERENCES:

<table>
<thead>
<tr>
<th></th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>CO2</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>CO3</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>CO4</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>CO5</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>CO6</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>3</td>
</tr>
</tbody>
</table>
OBJECTIVES:
- To gather knowledge on quality management, documentation and controlling for software quality.
- To provide knowledge on standards, models and tools used for quality management.
- To perform measurement and assessment of software quality.
- To introduce the basics and necessity of software testing.
- To introduce various testing techniques along with software production.

UNIT I INTRODUCTION TO SOFTWARE QUALITY & ARCHITECTURE 9

Suggested Activities:
- External learning - Software quality models.
- Report on quality plans.

Suggested Evaluation Methods:
- Assignment on quality models and quality plans.
- Evaluation of report.

UNIT II SQA COMPONENTS AND PROJECT LIFE CYCLE 9

Suggested Activities:
- Discussion on software quality assurance components.
- External learning - Quality assurance tools.

Suggested Evaluation Methods:
- Quiz on software quality assurance components.
- Assignment on quality assurance tools.

UNIT III SOFTWARE QUALITY INFRASTRUCTURE 9

Suggested Activities:
- Discussion on configuration management audit.
- Discussion on documentation control.
Suggested Evaluation Methods:
- Assignment on configuration management audit report.
- Quizzes on templates and checklist preparation.
- Quiz on documentation control.

UNIT IV SOFTWARE QUALITY MANAGEMENT, METRICS & STANDARDS

Suggested Activities:
- Discussion on ISO quality standards.
- External learning - Software quality metrics.

Suggested Evaluation Methods:
- Assignment on ISO quality standards.
- Quiz on process and product metrics.

UNIT V SOFTWARE TESTING

Suggested Activities:
- Discussion on test case generation and testing methods.

Suggested Evaluation Methods:
- Assignment on test case generation tools.
- Quiz on testing procedures.

OUTCOMES:
On completion of the course, the students will be able to:
1. Learn document control and manage software quality with the aid of tools and standards.
2. Distinguish between various software quality models.
3. Measure and assess software quality through process and product metrics.
4. Distinguish between the software quality standards.
5. Perform automated testing using test tools.
6. Document the testing procedures.

REFERENCES:

<table>
<thead>
<tr>
<th></th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>CO2</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>CO3</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>CO4</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>CO5</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>CO6</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>1</td>
</tr>
</tbody>
</table>

CA5022 INTRODUCTION TO SOCIAL NETWORK ANALYSIS

OBJECTIVES:

- To gain knowledge about the empirical and theoretical study of social networks, its structure and social network data sources.
- To study about the semantic technologies for social network analysis.
- To gain knowledge on visualization of social networks and its applications.
- To gain knowledge about social network analysis software for characterizing the network structure.
- To engage in critical thinking regarding the applicability of social network theory to various sociological phenomena.

UNIT I INTRODUCTION


Suggested Activities:

- Practical - Study of existing social networks and calculate the social network related metrics.
- Flipped classroom on fundamental mathematical knowledge on graphs and tutorial activity.
- External learning - Problems on calculation of ties, density, path, length, distance, betweenness, centrality, clique.
Suggested Evaluation Methods:
- Demonstration of social network creation and calculating the related metrics.
- Tutorial - Graphs and its related terminologies.
- Assignments on calculation of ties, density, path, length, distance, betweenness, centrality, clique.

UNIT II SOCIAL NETWORK ANALYSIS
9
Introduction to Social Networks Profiles – Types of Commercial Social Network Profiles (CSNP) – Quantitative and Qualitative Analysis of CSNP – Analysis of Social Networks Extracted from Log Files – Data Mining Methods Related to SNA and Log Mining – Clustering Techniques – Case Study.

Suggested Activities:
- Practical - Analysis of social network dataset.
- Flipped classroom on emerging applications of data mining based social network analysis techniques.
- External learning - Case study related to SNA.

Suggested Evaluation Methods:
- Demonstration of the analysis of social network log dataset.
- Tutorials on data mining applications.
- Assignments on data mining on SNA.

UNIT III SEMANTIC TECHNOLOGY FOR SOCIAL NETWORK ANALYSIS 9
Introduction to Ontology based Knowledge Representation – Ontology Languages for the Semantic Web – RDF and OWL – Modeling Social Network Data – Network Data Representation, Ontological Representation of Social Individuals and Relationships – Aggregating and Reasoning with Social Network Data – Advanced Representations.

Suggested Activities:
- Practical - Use of the features available in various ontology tools like Protégé.
- Flipped classroom on basic concepts of semantic web and ontology.
- External learning - Knowledge on semantic technology.

Suggested Evaluation Methods:
- Demonstration of created ontology.
- Tutorials - Semantic web related terminologies.
- Quizzes on semantic technology for SNA.

UNIT IV SOCIAL NETWORK MINING 9

Suggested Activities:
- Practical - Detection and mining of communities using various tools.
- Flipped classroom on basic concepts of online social networks (OSNs) and social network mining algorithms.
- External learning - Practical problems related to evaluation of community metrics.
Suggested Evaluation Methods:
- Demonstration - Community creation and mining.
- Tutorials on Social Network Mining.
- Assignments on community detection methods.

UNIT V VISUALIZATION AND APPLICATIONS OF SOCIAL NETWORKS

Visualization of Social Networks

Suggested Activities:
- Practical - Knowledge about tools related to social networks and implementation of social network visualizations using tools such as Gephi, Cytoscape.
- Flipped classroom on applications of social networks.
- External learning - How visualization is used in various real time SN applications.

Suggested Evaluation Methods:
- Demonstration of visual social networks
- Tutorials on applications of social networks.
- Quizzes on types of visualizations for social networks
- Group discussion on privacy and security of Aadhar.

TOTAL: 45 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
1. Understand basic principles behind network analysis algorithms and develop practical skills of network analysis.
2. Model and represent knowledge for social semantic Web.
3. Apply data mining techniques on social networks.
4. Use extraction and mining tools for analyzing Social networks.
5. Develop secure social network applications.

REFERENCES:
OBJECTIVES:
- To understand smart objects and IoT Architectures.
- To learn about various IoT related protocols.
- To build simple IoT systems using open hardware such as Arduino and Raspberry Pi.
- To understand data analytics and cloud in the context of IoT.
- To build IoT based smart systems.

UNIT I  FUNDAMENTALS OF IoT

Suggested Activities:
- Survey of different real world IoT applications.
- Assignments on operational principles of sensors and actuators.
- Mini project on building a smart system - Identify the sensors required for the system, connect sensors (such as temperature, pressure, light) to a suitable IoT hardware platform and take measurements.

Suggested Evaluation Methods:
- Evaluation of survey for breadth and depth - pair-wise comparison.
- Quiz on sensors and actuators.
- Demonstration of practical setup on connecting sensors.

UNIT II  IoT PROTOCOLS - I
Suggested Activities:
- Assignment on access technologies (simulator could be used).
- Flipped classroom on 6LoWPAN.
- Mini project on building a smart system - Choose appropriate access technology and connect the hardware to the Internet.

Suggested Evaluation Methods:
- Quiz on access technologies.
- Quiz on 6LoWPAN.
- Demonstration of practical setup on connecting to the Internet.

UNIT III  IoT PROTOCOLS - II
Routing over Low Power and Lossy Networks (RPL) – Application Transport Methods: Application Layer Not Present, Supervisory Control and Data Acquisition (SCADA) – Application Layer Protocols: CoAP and MQTT – Service discovery – mDNS.

Suggested Activities:
- Assignment on RPL (simulator could be used).
- Mini project on building a smart system - Choose appropriate application protocol and connect to the cloud using available open platforms (such as IBM Bluemix).

Suggested Evaluation Methods:
- Quiz on RPL for different topologies.
- Demonstration of practical setup on connecting to the cloud.

UNIT IV  CLOUD, FOG AND DATA ANALYTICS FRAMEWORKS

Suggested Activities:
- Use a simulator such as FogSim to study the characteristics of fog computing.
- Mini project on building a smart system - Choose appropriate analytics mechanisms to analyze the data collected, and build the application.

Suggested Evaluation Methods:
- Quiz on fog characteristics.
- Demonstration of application with analytics.

UNIT V  APPLICATIONS
Smart and Connected Cities: Street Layer, City Layer, Data Center Layer and Services Layer, Street Lighting, Smart Parking Architecture and Smart Traffic Control – Smart Transportation – Connected Cars.

Suggested Activities:
- Design the architecture and use cases for various smart systems (eg., agriculture, home automation, smart campus, smart hostel).
- Mini project on building a smart system - Enhance the system with additional smart features.
Suggested Evaluation Methods:
- Report and presentation of architecture solutions.
- Demonstration of complete smart system.

TOTAL: 45 PERIODS

OUTCOMES:
On completion of the course, the student will be able to:
1. Explain the concept and architecture of IoT.
2. Choose the right sensors and actuators for an application.
3. Analyze various protocols for IoT.
4. Apply data analytics and use cloud/fog offerings related to IoT.
5. Analyze applications of IoT in real time scenario.
6. Design an IoT based smart system using open hardware platforms and open cloud offerings.

REFERENCES:
7. https://www.arduino.cc/

<table>
<thead>
<tr>
<th></th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>CO2</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>CO3</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>CO4</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>CO5</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>CO6</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
</tbody>
</table>
OBJECTIVES:

- To understand the object oriented concepts and models used to analyze and design a system.
- To learn system modeling based on the requirements.
- To know about the design patterns, the object oriented approach to system development, modeling objects, relationships and interactions.
- To provide the knowledge of converting design to code for implementing an application using tools.
- To understand more patterns to design an application using tools and to implement it in real time.

UNIT I INTRODUCTION


Suggested Activities:

- Flipped classroom on preparing use cases, actors involved in the ATM system.
- Flipped classroom on preparing activity diagram and state machine diagram for telephone line.

Suggested Evaluation Methods:

- Quizzes on different concepts of object oriented systems.
- Assignment on UML diagrams for library management system using StarUML tool.

UNIT II REQUIREMENTS AND MODELING


Suggested Activities:

- Flipped classroom on gathering requirements for library management system.
- Assignment on preparing SSD for NextGen POS case study.
- Assignment on operation contracts for process sale scenario.

Suggested Evaluation Methods:

- Quizzes on requirement types.
- Assessment on domain model and SSDs for Monopoly game.

UNIT III DESIGN PATTERNS

Suggested Activities:
- External learning - Use of open source UML tools like BOUML, Papyrus, Umbrello to perform modeling.
- Flipped classroom on consider the video store, list the classes and objects and assign the responsibilities for each class.

Suggested Evaluation Methods:
- Assignment on design patterns.
- Quizzes on logical architecture.

UNIT IV MAPPING DESIGN TO CODE
Mapping Designs to Code: Class Definitions from DCDs, Methods from Interaction Diagrams, Collection Classes, Exceptions and Error Handling – Test Driven Development and Refactoring – UML Tools and UML as Blueprint.

Suggested Activities:
- External learning - Use BOUML, Eclipse, UModel tools to generate code from diagram.
- External learning - Use TOOTSIE, Object Tester tools to test the code.

Suggested Evaluation Methods:
- Assignment on carrying out simple code generation using code generation tools.
- Assessment on testing using various object oriented testing tools.

UNIT V MORE PATTERNS AND CASE STUDY

Suggested Activities:
- Flipped classroom on how to store and retrieve objects, use different storage mechanisms and formats such as RDBs, records in flat files or XML in files.
- Flipped classroom on the preparation of the software document architecture and technical memo for NextGen POS.
- Assignment on mapping of an object to a record on relational database schema.

Suggested Evaluation Methods:
- Quizzes on various patterns.
- Assessment on object storing and retrieving in RDBs.

TOTAL: 45 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
1. Familiar with object oriented paradigm, concepts and development processes.
2. Develop models based on requirements and explore the conceptual model into various scenarios and applications.
3. Familiar with the design patterns and mapping design to code.
4. Able to use the OO testing techniques to test the various applications.
5. Able to use the graphical UML representation using tools, such as Rational Rose, StarUML.
6. Document the architecture and apply the concepts of architectural design for deploying the code for software.
REFERENCES:

<table>
<thead>
<tr>
<th></th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>CO2</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>CO3</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>CO4</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>CO5</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>CO6</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>

CA5025 ARTIFICIAL INTELLIGENCE L T P C
3 0 0 3

OBJECTIVES:
- To know the underlying structure behind intelligence mathematically.
- To know the logical implications in computational intelligence.
- To know the automated learning techniques.
- To study the techniques of knowledge representation.
- To explore artificial intelligence techniques in real – time scenarios.

UNIT I INTELLIGENT AGENTS AND KNOWLEDGE REPRESENTATION 9

Suggested Activities:
- Flipped classroom on structure of agents.
- Solving exercise questions.
- Examples of knowledge representation through different methods and reasoning.
- Practical - Ontology creation using Protégé.

Suggested Evaluation Methods:
- Tutorial problems on various topics of the unit.
- Assignment problems on knowledge representation.
- Quizzes on agents, environments and search.
- Evaluation of the programming exercises.
UNIT II  SEARCH TECHNIQUES  9

Suggested Activities:
- Flipped classroom on uninformed search - searching with costs.
- In-class activity on solving puzzles with uninformed and informed searches.
- Practical - Implementation of search through Python/other languages.

Suggested Evaluation Methods:
- Tutorial problems on various topics of the unit.
- Assignments on puzzles with uninformed and informed searches.
- Quizzes on environments and search
- Evaluation of the programming exercises.

UNIT III  REASONING WITH LOWER ORDER LOGICS  9

Suggested Activities:
- Reasoning methods through puzzles and real life scenarios.
- Practical - Inference through Prolog/Python.
- Practical - Programming through Prolog/ Python for various topics such as reasoning through resolution.

Suggested Evaluation Methods:
- Tutorials on reasoning methods.
- Assignments on different topics of the unit.
- Quizzes on inference techniques in logic.
- Evaluation of the programming exercises.

UNIT IV  ARTIFICIAL INTELLIGENCE PLANNING  9
Classical Planning – Partial Order Planning – Graph Plan and SAT Plan – Hierarchical Planning – Planning and Acting in Nondeterministic Domains – Multi Agent Planning.

Suggested Activities:
- Flipped classroom on planning types and the background of plan.
- Out of class activity - Classical planning, Boolean satisfiability.
- In-class activity - Graph plan.
- Programming using PDDL/ python.

Suggested Evaluation Methods:
- Tutorials - Planning types and the background of plan.
- Assignments on graph plan.
- Quizzes on planning basics.
- Evaluation of the programming exercise.

UNIT V  LEARNING TECHNIQUES  9
Suggested Activities:
- Flipped classroom on knowledge in learning.
- Assignments on problem solving in learning techniques.
- Practical - Programming exercises using Python/other programming languages such as: Programming for HMM.
- Explore an available Chatbot model such as Watson and adapt to a specific domain such as Education or Customer relations.

Suggested Evaluation Methods:
- Tutorials and quizzes on knowledge in learning.
- Evaluation of the programming exercise.

TOTAL: 45 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
1. Understand the search techniques.
2. Apply the search techniques to real-time problems.
3. Apply the reasoning techniques to real world problems.
4. Understand the representation of knowledge.
5. Understand the learning techniques.
6. Apply AI techniques in developing real world applications.

REFERENCES:

<table>
<thead>
<tr>
<th></th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>CO2</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>CO3</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>CO4</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>CO5</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>CO6</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>2</td>
</tr>
</tbody>
</table>
OBJECTIVES:
- To know the mathematical basis of computer graphics.
- To acquire knowledge in computer graphics modeling, animation, and rendering.
- To create graphical applications.
- To acquire knowledge about tools and technologies related to graphics.
- To create visually realistic animations.

UNIT I INTRODUCTION TO COMPUTER GRAPHICS
9

Suggested Activities:
- Flipped classroom on basic vector and arithmetic operations on vector.
- Practical - Use OpenGL to create visual objects using lines and apply clipping algorithms.
- Assignments on line drawing algorithms and clipping algorithms.

Suggested Evaluation Methods:
- Tutorials on arithmetic operations on vector.
- Demonstration of line drawing algorithms and applying clipping algorithms over the created objects.
- Assignments on solving problems on vectors.

UNIT II MODELING AND TRANSFORMATIONS OF OBJECTS
9

Suggested Activities:
- Practical - Creating three dimensional solid objects and apply transformations.
- Brainstorming session on different modeling techniques.
- Assignments on three dimensional transformations.

Suggested Evaluation Methods:
- Demonstration on creation of three dimensional solid objects and applying various transformations.
- Creativity and production of impressive visual imagery.
- Quizzes on various topics of the unit.

UNIT III VIEWING AND VISUAL REALISM
9
Suggested Activities:
- Assignments on hidden surface removal methods.
- Practical - Adding shadows and lighting effects on modeled objects.

Suggested Evaluation Methods:
- Tutorials on hidden surface removal algorithms.
- Demonstrations by creating visually aesthetic scenes.

UNIT IV SURFACE DESIGN

Suggested Activities:
- Assignments on generating different types of curves.
- Practical - Drawing curves and curved objects.

Suggested Evaluation Methods:
- Demonstration on drawing curves and curved objects.
- Creativity and production of impressive three dimensional objects.
- Quizzes on color theory, ray tracing process and other topics of the unit.

UNIT V ANIMATIONS

Suggested Activities:
- Practical - To create realistic animations.
- External learning - The process of animation movie making.

Suggested Evaluation Methods:
- Demonstration on animated movies created using various techniques.
- Quizzes on animation movie making.

TOTAL: 45 PERIODS

OUTCOMES:
On completion of the course, the students will be able to,
1. Articulate the concepts and techniques used in three-dimensional graphics.
2. Understand and implement algorithms related to graphics creation.
3. Design and model graphical structures.
4. Understand and comprehend the graphical algorithms.
5. Design visually realistic graphical applications.
6. Design and develop simple and realistic animations.

REFERENCES:

<table>
<thead>
<tr>
<th></th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>CO2</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>CO3</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>CO4</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>CO5</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>CO6</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>

CA5027 HUMAN COMPUTER INTERACTION L T P C 3 0 0 3

OBJECTIVES:
• To learn the principles and fundamentals of HCI.
• To understand components of interfaces and screens, including windows, menus and controls.
• To understand user interface design principles and apply them to designing an interface.
• To learn user interface designs through usability inspection and user models.
• To understand the rationale and guidelines for an effective interface design methodology.

UNIT I DESIGN PROCESS

Suggested Activities:
• Flipped classroom on basic knowledge on the HCI design process.
• External learning - Exploration of various human computer interfaces.
• Practical - Implementation of a simple user interface using tools like scratch, React, Adobe XD.

Suggested Evaluation Methods:
• Tutorials on HCI design process.
• Assignment on comparison of various interfaces.
• Demonstration of created user interfaces for web applications and other customized applications.
UNIT II  DESIGN AND EVALUATION OF INTERACTIVE SYSTEMS


Suggested Activities:
- Flipped classroom on designing a good User Interface system based on design rules.
- External learning - Techniques related to evaluation of HCI design.
- Practical - Development and validation of user interfaces using various evaluation techniques.

Suggested Evaluation Methods:
- Tutorials on usage of design rules to create interfaces.
- Assignment on applying evaluation techniques on different user interfaces.
- Demonstration of interface design and evaluation.

UNIT III  COMMUNICATION MODELS


Suggested Activities:
- Flipped classroom on basic knowledge of various models used in HCI design.
- External learning - Design and implementation of various models used in HCI design.
- Practical - Implementation of design rules to execute the models.

Suggested Evaluation Methods:
- Tutorials on task models.
- Assignment on dialog models and task models.
- Demonstration on model based user interface design.

UNIT IV  EXPERIMENTAL DESIGN AND STATISTICAL ANALYSIS OF HCI


Suggested Activities:
- Flipped classroom on basic concepts of probability and statistics.
- External learning - Problems related to hypothesis testing.
- Practical - Implementation of UI and analyzing using various statistical measures.

Suggested Evaluation Methods:
- Tutorials on statistical testing related to UI evaluation parameters.
- Assignments on hypothesis testing for UI parameters.
- Demonstration of UI development and statistical testing.
UNIT V  DIALOGUE AND CURRENT TRENDS


Suggested Activities:
- Flipped classroom on basic concepts of dialogue notations and design.
- External learning - Study of how virtual reality interface are used in various real time applications.
- Practical - Implementation of virtual reality based visualization for dialogue based systems.

Suggested Evaluation Methods:
- Tutorials on recent trends in human computer interface systems.
- Assignment on dialogue notation representation for various interfaces.
- Demonstration on virtual reality based application interface.

TOTAL: 45 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
1. Interpret the contributions of human factors and technical constraints on human computer interaction.
2. Evaluate the role of current HCI theories in the design of software.
3. Design and develop interfaces related to real applications.
4. Apply exploratory and experimental research methods in HCI.
5. Be equipped with principles and guidelines of user centered interface design process, evaluation methodologies and tools to analyze the interfaces.
6. Implement human computer interfaces for different applications using various tools and technologies.

REFERENCES:
OBJECTIVES:
- To learn about the physical layer and MAC layer of WSNs.
- To understand the data centric computing to be followed in WSNs.
- To study about the routing protocols followed in WSNs.
- To study about data aggregation and in-network processing.
- To explore various motes, sensor network operating systems, databases and development platforms.

UNIT I  FUNDAMENTALS OF WSN  9

Suggested Activities:
- External learning - Exploring various sensors, the corresponding actuators, various motes and their configuration (sensors supported, microcontroller and the clock speed, Flash, RAM, Battery capacity, RF transceivers and data rate supported).
- Flipped classroom on accuracy, hysteresis and resolution of sensors.
- Assignments on calculations of energy requirement for transmission, receiving and channel sensing.

Suggested Evaluation Methods:
- Assignments on various types of sensors, actuators and motes.
- Quiz and discussion on accuracy, hysteresis and resolution of sensors.
- Assignments on problems related to energy consumption in WSNs.

UNIT II  MAC LAYER OF WSN AND ZIGBEE STANDARD  9
Suggested Activities:
- External learning - A study of Wireless HART, 6LoWPAN and ISA 100.11a standards.
- Flipped classroom on different roles of nodes in WSNs and different types of ZigBee devices.
- Analyzing duty cycle and sleep cycle of S-MAC protocol.

Suggested Evaluation Methods:
- Assignments on various standards available for WSNs.
- Quiz and discussion on roles of nodes and different types of ZigBee devices.
- Assignments on problems related to duty cycle of S-MAC protocol.

UNIT III  DATA CENTRIC COMPUTING IN WSN


Suggested Activities:
- Flipped classroom on data centric computing and information centric networks.
- Assignments on analyzing the generation and consumption of energy with nonconventional energy sources.
- External learning - Sensor network platforms and tools and sensor network databases.

Suggested Evaluation Methods:
- Quiz and discussion on data centric computing and information centric networks.
- Assignments on problems regarding generation and consumption of energy sources.
- Assignments on sensor network platforms, tools and sensor network databases.

UNIT IV  SYNCHRONIZATION, LOCALIZATION AND TRACKING IN WSNs


Suggested Activities:
- External learning - Exploring tracking of objects using ultrasonic sensors and camera nodes.
- Exploring the idea of smart cities using Object Tracking Sensor Networks (OTSN).
- Flipped classroom on scene analysis, GPS, RFID and location based services.

Suggested Evaluation Methods:
- Assignments on tracking of objects using ultrasonic sensors and camera nodes.
- Assignments on designing WSNs to locate and track moving objects using ultrasonic sensors or camera nodes for smart cities.
- Quiz and discussion on scene analysis, GPS, RFID and location based services.

UNIT V  SENSOR NETWORK PLATFORMS AND TOOLS

Suggested Activities:
- Practical - Exploring various network simulators available to carry out experiments in WSNs and various WSN testbeds: WISBED, SensLAB, MoteLAB, CitySense and Sensei.
- Flipped classroom on Contiki OS and COOJA IDE.
- Assignments on developing Arduino sketches and WSN simulation in NS3.

Suggested Evaluation Methods:
- Assignments on WSN simulators and WSN testbeds.
- Quiz and discussion on Contiki OS and COOJA IDE.
- Assignments on writing Arduino sketches for socially relevant projects and creating a sensor network topology in ns – 2.35 with Mannasim patch or in NS3.

TOTAL: 45 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
1. Understand different types of sensors, their actuators and the architecture of motes.
2. Design the topology of WSNs using different types of ZigBee devices and understanding their roles.
3. Understand and apply data centric computing in wireless sensor networks.
4. Apply appropriate localization techniques for different scenarios.
5. Manage sensor networks by synchronizing the time, locating and tracking objects.
6. Carry out experiments in simulators and real sensors.

REFERENCES:

<table>
<thead>
<tr>
<th></th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>CO2</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>CO3</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>CO4</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>CO5</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>CO6</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>3</td>
</tr>
</tbody>
</table>
OBJECTIVES:

- To learn the fundamentals of 5G internet.
- To understand the concept of small cells in 5G mobile networks.
- To learn the mobile clouds in 5G network context.
- To understand the role of cognitive radios in 5G networks.
- To learn the security issues in 5G networks.

UNIT I PERVASIVE CONNECTED WORLD AND 5G INTERNET


Suggested Activities:

- Flipped classroom on Ten Pillars of 5G.
- Assignment on millimeter wave mobile communication.
- External learning - 5G in global level.

Suggested Evaluation Methods:

- Viva Voce on assignment topic.
- Group discussion on different generations of telecommunication networks.
- Quizzes on spectrum allocation strategies for 5G.

UNIT II SMALL CELLS FOR 5G MOBILE NETWORKS


Suggested Activities:

- Flipped classroom on the types of small cells.
- Assignment on issues in femtocells.
- External learning – Small cell challenges.

Suggested Evaluation Methods:

- Viva voce on assignment topic.
- Quiz on the drawbacks of dense deployment of Wi-Fi systems.

UNIT III COOPERATION FOR NEXT GENERATION WIRELESS NETWORKS


Suggested Activities:

- Flipped classroom on network coding.
- External learning – Cooperative MAC protocols.
- Assignment on packet exchange in PRCSMA.
Suggested Evaluation Methods:
- Viva voce on assignment topic.
- Quiz on NCCARQ operation under realistic channel conditions.
- Practical - Assessing the performance of NC-aided MAC protocols in event-driven C++ simulator.

UNIT IV MOBILE CLOUDS AND COGNITIVE RADIO 9

Suggested Activities:
- External learning - Network coding.
- Assignment on spectrum optimization using cognitive radio.
- External learning – Key requirements and challenges for 5G cognitive terminals.
- Assignment on component of a cognitive radio terminal.

Suggested Evaluation Methods:
- Viva voce on assignment topics.
- Quiz on carrier aggregation.

UNIT V SECURITY AND SELF ORGANISING NETWORKS 9

Suggested Activities:
- External learning - 5G communications system architecture.
- Flipped classroom on security issues and challenges in communication systems.
- Assignment on centralised 5G mobile botnet.

Suggested Evaluation Methods:
- Viva voce on assignment topics.
- Quiz on D-SON and C-SON architectures.
- Group discussion on Attacks on 4G Access Network.

TOTAL: 45 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
- Compare the 5G network with older generations of networks.
- Identify suitable small cells for different applications in 5G networks.
- Simulate 5G network scenarios.
- Connect applications to mobile cloud.
- Design applications with 5G network support.
- Analyze the security risks in 5G networks.
REFERENCES:

OBJECTIVES:
- To provide an understanding of computer forensics fundamentals.
- To analyze various computer forensics technologies.
- To identify methods for data recovery.
- To apply the methods for preservation of digital evidence.
- To learn about the types of attacks and remedial actions in the context of systems, networks, images and videos.

UNIT I INCIDENT AND INCIDENT RESPONSE
9

Suggested Activities:
- External learning - Survey of forensics tools such as WinHex, EnCase, FTK, or ProDiscover.
- Practical - Demonstrate some of the mechanisms used by malicious attackers as well as forensic experts to disrupt computer networks and manipulate information access.

Suggested Evaluation Methods:
- Assignments on steps of incident response methodology.
- Quizzes on various security mechanisms.
UNIT II  FILE STORAGE AND DATA RECOVERY

Suggested Activities:
- External learning - Survey of various tools for data storage and access, bypassing filtered [blocked] ports, reviewing Internet activity, Open-source forensic tools for file storage and data recovery.

Suggested Evaluation Methods:
- Assignments on reconstruction of past events.
- Quizzes on different types of file systems.

UNIT III  NETWORK AND EMAIL FORENSICS

Suggested Activities:
- Practical - Familiarizing with Port Redirection tools: Quick ‘n Easy FTP Server, FPIPE and FPORT.
- Practical - Study of the forensics tools.

Suggested Evaluation Methods:
- Demonstration of the practical implementations.
- Real time problems on network data analysis.

UNIT IV  SYSTEM FORENSICS

Suggested Activities:
- Demonstration on MD5Hash tool.
- Practical - IE Activity analysis.

Suggested Evaluation Methods:
- Assignments on evidence collection.
- Quizzes on ethical issues and live system investigation.

UNIT V  IMAGE AND VIDEO FORENSICS

Suggested Activities:
- External learning - Steganography.
- Practical - Steganalysis tool.
Suggested Evaluation Methods:
- Assignments on data compression.
- Quizzes on file formats and copyright issues.

TOTAL: 45 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
1. Recognize attacks on systems.
2. Design a counter attack incident response and incident-response methodology.
3. Illustrate the methods for data recovery, evidence collection and data seizure.
4. Understand network and email attacks and forensic investigation with tools.
5. Use forensic tools and collect evidences of a computer crime.
6. Analyze various image encryption/decryption, steganography and fraud in image.

REFERENCES:

<table>
<thead>
<tr>
<th>CO1</th>
<th>CO2</th>
<th>CO3</th>
<th>CO4</th>
<th>CO5</th>
<th>CO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>PO1</td>
<td>PO2</td>
<td>PO3</td>
<td>PO4</td>
<td>PO5</td>
<td>PO6</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
</tbody>
</table>

CA5031 NETWORK PROGRAMMING AND MANAGEMENT L T P C 3 0 0 3

OBJECTIVES:
- To learn the basics of socket programming using TCP Sockets.
- To learn about socket options.
- To explore the features of raw sockets.
- To learn and develop macros for including objects in MIB structure.
- To have knowledge on various network management tools.
UNIT I        SOCKETS AND APPLICATION DEVELOPMENT


Suggested Activities:
- Assignment on Syntax and interpretation of various Socket Programming System Calls.
- Practical - Implement basic socket programs using C.

Suggested Evaluation Methods:
- Quiz on system calls.
- Evaluation of the implemented programs with appropriate test cases.

UNIT II         SOCKET OPTIONS


Suggested Activities:
- Assignment on socket options implementing using C for specific scenarios.
- Practical - Implementation of DNS operation with gethostbyname, gethostbyaddr, getservbyname and getservbyport functions using C.
- Practical - Implementation of protocol independent functions in C.

Suggested Evaluation Methods:
- Testing for the respective socket option's role in the scenario chosen.
- Quiz on roles of various protocols dependent and independent functions.

UNIT III        ADVANCED SOCKETS


Suggested Activities:
- Assignments on IPv4 and IPv6.
- Practical - Programs using Pthread.
- Practical - Implementation of program in C for handling raw socket.

Suggested Evaluation Methods:
- Quiz on IPv4 and IPv6 interoperability.
- Testing the program implemented using raw sockets.

UNIT IV        SIMPLE NETWORK MANAGEMENT

Suggested Activities:
- Assignment on SNMP architecture and features of versions.
- Assignment to develop macros for new objects in MIB.

Suggested Evaluation Methods:
- Quiz on SNMP versions.
- Test for the correct definition of the access rights for the MIB objects.

UNIT V NETWORK MANAGEMENT TOOLS & SYSTEMS


Suggested Activities:
- Practical - Examine the headers and contents of IP using tcpdump or Wireshark.
- Practical - Using suitable network monitoring tool, analyze the traffic conditions in TCP/IP network.
- Practical - Analyze the network performance (delay) using appropriate system utilities.

Suggested Evaluation Methods:
- Verification of header and contents.
- Performance evaluation for various scenarios.

TOTAL: 45 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
1. Implement client/server communications using TCP and UDP Sockets.
2. Describe the usage of socket options for handling various Sockets in programming.
3. Understand handling of raw sockets.
4. Explain functionalities of SNMP and MIB structure.
5. Experiment with various tools available to manage a network.
6. Handle technical issues in a network.

REFERENCES:
OBJECTIVES:
- To learn the fundamentals of semantic web and to conceptualize and depict ontology for semantic web.
- To make a study of languages for semantic web.
- To learn about the ontology learning algorithms and to utilize in the development of an application.
- To know the fundamental concepts of ontology management.
- To learn the applications related to semantic web.

UNIT I  THE QUEST FOR SEMANTICS

Suggested Activities:
- Flipped classroom on semantic web background and tutorial activity.
- Brainstorming session - Various knowledge representation formats.
- Practical - Design of simple ontology on their domain of interest using tools like Protégé.
- Practical - Installing EasyRdf in the system and including this in PHP (EasyRdf is a PHP library, which can be used to consume and produce RDF).

Suggested Evaluation Methods:
- Tutorials on semantic web basics.
- Quizzes on knowledge representation formats
- Demonstration of simple implemented ontology.
UNIT II  LANGUAGES FOR SEMANTIC WEB AND ONTOLOGIES

Suggested Activities:
- Flipped classroom on comparison of various semantic web related languages and tutorial.
- Practical - Creation of RDF documents.
- Practical - Use of OWL language to represent relationships, properties and to provide inferences from created ontology.

Suggested Evaluation Methods:
- Quizzes on various ontology related languages
- Demonstration of knowledge inference from created ontologies.

UNIT III  ONTOLOGY LEARNING FOR SEMANTIC WEB

Suggested Activities:
- Flipped classroom on natural language processing techniques like statistical text analysis, term extraction, word sense disambiguation, concept extraction and tutorial activity.
- Practical - Term extraction and term disambiguation from corpus using Alchemy like API.
- Extended Reading from the site - https://nlp.stanford.edu/fsnlp/.

Suggested Evaluation Methods:
- Tutorials on language processing techniques.
- Demonstration on term extraction and term disambiguation.

UNIT IV  ONTOLOGY MANAGEMENT AND TOOLS

Suggested Activities:
- Flipped classroom on study of various ontology related tools.
- Practical - Use of any tool to apply SPARQL queries and implement reasoning for avoiding inconsistencies
- Practical - Merging two ontologies, applying association rules, applying clustering algorithms

Suggested Evaluation Methods:
- Tutorials on ontology related tools like Protege, Ontolingua, Webonto.
- Demonstration of clustering, merging ontologies and Sparql queries.
UNIT V  APPLICATIONS

Suggested Activities:
- Flipped classroom on other applications of semantic web.
- Practical - Simple application like chat bot, semantic search engine creation using topic map data models extracted from Ontopia/Mappa.
- Practical - Creating intelligent expert systems using semantic Wikis like SMW+.

Suggested Evaluation Methods:
- Quizzes on semantic web applications
- Demonstration of applications created using tools.

TOTAL: 45 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
1. Create ontology for a given domain.
2. Develop an application using ontology languages and tools.
3. Understand the concepts of semantic web.
4. Use ontology related tools and technologies for application creation.
5. Design and develop applications using semantic web.
6. Understand the standards related to semantic web.

REFERENCES:

<table>
<thead>
<tr>
<th></th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>CO2</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>CO3</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>CO4</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>CO5</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>CO6</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
</tr>
</tbody>
</table>
OBJECTIVES:

- To gain knowledge of soft computing theories and its fundamentals.
- To design a soft computing system required to address a computational task.
- To learn and apply artificial neural networks, fuzzy sets and fuzzy logic and genetic algorithms in problem solving and use of heuristics based on human experience.
- To introduce the ideas of fuzzy sets, fuzzy logic and to become familiar with neural networks that can learn from available examples and generalize to form appropriate rules for inferencing systems.
- To familiarize with genetic algorithms and other random search procedures while seeking global optimum in self-learning situations.

UNIT I  FUZZY COMPUTING  9

Suggested Activities:
- Install MatLab Fuzzy Logic Toolbox and ANN toolbox to design and simulate systems.

Suggested Evaluation Methods:
- Quiz – basic concepts of fuzzy logic and operations.

UNIT II  FUNDAMENTALS OF NEURAL NETWORKS  9

Suggested Activities:
- Develop a supervised model to Train neural net that uses the AND/OR/XOR two input binary/bipolar input and output data and learn linear models to understand the importance of initialization parameters.

Suggested Evaluation Methods:
- Project Demonstration.
- Implementation Evaluation with new input set.

UNIT III  BACKPROPAGATION NETWORKS  9

Suggested Activities:
- Develop a supervised model to
  - Train neural net that uses the XOR three input binary/bipolar input and output data and learn linear models to understand the importance of learning parameters.
  - Train a linear / nonlinear model with one hidden layer, two hidden layers etc.
- Observe the performance with different learning rates and draw the graph depicting the error rate with iterations.

**Suggested Evaluation Methods:**
- Project Demonstration.
- Implementation Evaluation with new input set.

**UNIT IV  COMPETETIVE NEURAL NETWORKS**


**Suggested Activities:**
Develop an unsupervised model to
- Train neural net that uses any Dataset and Plot the cluster of patterns.

**Suggested Evaluation Methods:**
- Project Demonstration.
- Implementation Evaluation with new input set.

**UNIT V  GENETIC ALGORITHM**


**Suggested Activities:**
- Implement GA for the travelling salesman problem to find the shortest path that visits all cities in a set exactly once.

**Suggested Evaluation Methods:**
- Implementation evaluation by testing the code on different route map and check the optimal solution.

**OUTCOMES:**
On completion of the course, the students will be able to:
1. Identify and describe soft computing techniques and their roles in building intelligent machines.
2. Recognize the feasibility of applying a soft computing methodology for a particular problem.
3. Apply fuzzy logic and reasoning to handle uncertainty and solve engineering problems.
4. Apply genetic algorithms to optimization problems.
5. Design neural networks to pattern classification and regression problems using soft computing approach.
6. Describe the importance of tolerance of imprecision and uncertainty to a design of robust and low cost intelligent machines.

**REFERENCES:**
OE5091 BUSINESS DATA ANALYTICS

OBJECTIVES:
- To understand the basics of business analytics and its life cycle.
- To gain knowledge about fundamental business analytics.
- To learn modeling for uncertainty and statistical inference.
- To understand analytics using Hadoop and Map Reduce frameworks.
- To acquire insight on other analytical frameworks.

UNIT I OVERVIEW OF BUSINESS ANALYTICS


Suggested Activities:
- Case studies on applications involving business analytics.
- Converting real time decision making problems into hypothesis.
- Group discussion on entrepreneurial opportunities in Business Analytics.

Suggested Evaluation Methods:
- Assignment on business scenario and business analytical life cycle process.
- Group presentation on big data applications with societal need.
- Quiz on case studies.
UNIT II  ESSENTIALS OF BUSINESS ANALYTICS


Suggested Activities:

- Solve numerical problems on basic statistics.
- Explore chart wizard in MS Excel Case using sample real time data for data visualization.
- Use R tool for data visualization.

Suggested Evaluation Methods:

- Assignment on descriptive analytics using benchmark data.
- Quiz on data visualization for univariate, bivariate data.

UNIT III  MODELING UNCERTAINTY AND STATISTICAL INFERENCE


Suggested Activities:

- Solving numerical problems in sampling, probability, probability distributions and hypothesis testing.
- Converting real time decision making problems into hypothesis.

Suggested Evaluation Methods:

- Assignments on hypothesis testing.
- Group presentation on real time applications involving data sampling and hypothesis testing.
- Quizzes on topics like sampling and probability.

UNIT IV  ANALYTICS USING HADOOP AND MAPREDUCE FRAMEWORK


Suggested Activities:

- Practical – Install and configure Hadoop.
- Practical – Use web based tools to monitor Hadoop setup.
- Practical – Design and develop MapReduce tasks for word count, searching involving text corpus etc.

Suggested Evaluation Methods:

- Evaluation of the practical implementations.
- Quizzes on topics like HDFS and extensions to MapReduce.
UNIT V OTHER DATA ANALYTICAL FRAMEWORKS

Overview of Application development Languages for Hadoop – PigLatin – Hive – Hive Query Language (HQL) – Introduction to Pentaho, JAQL – Introduction to Apache: Sqoop, Drill and Spark, Cloudera Impala – Introduction to NoSQL Databases – Hbase and MongoDB.

Suggested Activities:
- Practical – Installation of NoSQL database like MongoDB.
- Practical – Demonstration on Sharding in MongoDB.
- Practical – Install and run Pig
- Practical – Write PigLatin scripts to sort, group, join, project, and filter data.
- Design and develop algorithms to be executed in MapReduce involving numerical methods for analytics.

Suggested Evaluation Methods:
- Mini Project (Group) – Real time data collection, saving in NoSQL, implement analytical techniques using Map-Reduce Tasks and Result Projection.

TOTAL: 45 PERIODS

OUTCOMES:
On completion of the course, the student will be able to:
- Identify the real world business problems and model with analytical solutions.
- Solve analytical problem with relevant mathematics background knowledge.
- Convert any real world decision making problem to hypothesis and apply suitable statistical testing.
- Write and Demonstrate simple applications involving analytics using Hadoop and MapReduce
- Use open source frameworks for modeling and storing data.
- Apply suitable visualization technique using R for visualizing voluminous data.

REFERENCES:
OBJECTIVES:
- Summarize basics of industrial safety
- Describe fundamentals of maintenance engineering
- Explain wear and corrosion
- Illustrate fault tracing
- Identify preventive and periodic maintenance

UNIT I  INTRODUCTION  9
Accident, causes, types, results and control, mechanical and electrical hazards, types, causes and preventive steps/procedure, describe salient points of factories act 1948 for health and safety, wash rooms, drinking water layouts, light, cleanliness, fire, guarding, pressure vessels, etc, Safety color codes. Fire prevention and firefighting, equipment and methods.

UNIT II  FUNDAMENTALS OF MAINTENANCE ENGINEERING  9
Definition and aim of maintenance engineering, Primary and secondary functions and responsibility of maintenance department, Types of maintenance, Types and applications of tools used for maintenance, Maintenance cost & its relation with replacement economy, Service life of equipment.

UNIT III  WEAR AND CORROSION AND THEIR PREVENTION  9

UNIT IV  FAULT TRACING  9
Fault tracing-concept and importance, decision tree concept, need and applications, sequence of fault finding activities, show as decision tree, draw decision tree for problems in machine tools, hydraulic, pneumatic, automotive, thermal and electrical equipment’s like, i. Any one machine tool, ii. Pump iii. Air compressor, iv. Internal combustion engine, v. Boiler, vi. Electrical motors, Types of faults in machine tools and their general causes.
UNIT V PERIODIC AND PREVENTIVE MAINTENANCE

Periodic inspection-concept and need, degreasing, cleaning and repairing schemes, overhauling of mechanical components, overhauling of electrical motor, common troubles and remedies of electric motor, repair complexities and its use, definition, need, steps and advantages of preventive maintenance. Steps/procedure for periodic and preventive maintenance of: I. Machine tools, ii. Pumps, iii. Air compressors, iv. Diesel generating (DG) sets, Program and schedule of preventive maintenance of mechanical and electrical equipment, advantages of preventive maintenance. Repair cycle concept and importance

OUTCOMES:
Students will be able to:
CO1: Ability to summarize basics of industrial safety
CO2: Ability to describe fundamentals of maintenance engineering
CO3: Ability to explain wear and corrosion
CO4: Ability to illustrate fault tracing
CO5: Ability to identify preventive and periodic maintenance

TOTAL: 45 PERIODS

REFERENCES:

OE5093 OPERATIONS RESEARCH

OBJECTIVES:
- Solve linear programming problem and solve using graphical method.
- Solve LPP using simplex method
- Solve transportation, assignment problems
- Solve project management problems
- Solve scheduling problems

UNIT I LINEAR PROGRAMMING
Introduction to Operations Research – assumptions of linear programming problems - Formulations of linear programming problem – Graphical method
UNIT II ADVANCES IN LINEAR PROGRAMMING
Solutions to LPP using simplex algorithm - Revised simplex method - primal dual relationships – Dual simplex algorithm - Sensitivity analysis

UNIT III NETWORK ANALYSIS – I
Transportation problems -Northwest corner rule, least cost method, Voges’s approximation method - Assignment problem - Hungarian algorithm

UNIT IV NETWORK ANALYSIS – II
Shortest path problem: Dijkstra’s algorithm, Floyd’s algorithm, systematic method - CPM/PERT

UNIT V NETWORK ANALYSIS – III
Scheduling and sequencing - single server and multiple server models - deterministic inventory models - Probabilistic inventory control models

TOTAL: 45 PERIODS

OUTCOMES:
Students will be able to:
CO1: To formulate linear programming problem and solve using graphical method.
CO2: To solve LPP using simplex method
CO3: To formulate and solve transportation, assignment problems
CO4: To solve project management problems
CO5: To solve scheduling problems

<table>
<thead>
<tr>
<th></th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
<th>PO7</th>
<th>PO8</th>
<th>PO9</th>
<th>PO10</th>
<th>PO11</th>
<th>PO12</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CO2</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CO3</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CO4</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CO5</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

REFERENCES:
OBJECTIVES:

- Summarize the costing concepts and their role in decision making
- Infer the project management concepts and their various aspects in selection
- Interpret costing concepts with project execution
- Develop knowledge of costing techniques in service sector and various budgetary control techniques
- Illustrate with quantitative techniques in cost management

UNIT I INTRODUCTION TO COSTING CONCEPTS 9
Objectives of a Costing System; Cost concepts in decision-making; Relevant cost, Differential cost, Incremental cost and Opportunity cost; Creation of a Database for operational control.

UNIT II INTRODUCTION TO PROJECT MANAGEMENT 9
Project: meaning, Different types, why to manage, cost overruns centres, various stages of project execution: conception to commissioning. Project execution as conglomeration of technical and nontechnical activities, Detailed Engineering activities, Pre project execution main clearances and documents, Project team: Role of each member, Importance Project site: Data required with significance, Project contracts.

UNIT III PROJECT EXECUTION AND COSTING CONCEPTS 9
Project execution Project cost control, Bar charts and Network diagram, Project commissioning: mechanical and process, Cost Behavior and Profit Planning Marginal Costing; Distinction between Marginal Costing and Absorption Costing; Break-even Analysis, Cost-Volume-Profit Analysis, Various decision-making problems, Pricing strategies: Pareto Analysis, Target costing, Life Cycle Costing.

UNIT IV COSTING OF SERVICE SECTOR AND BUDGETERY CONTROL 9
Just-in-time approach, Material Requirement Planning, Enterprise Resource Planning, Activity-Based Cost Management, Bench Marking; Balanced Score Card and Value-Chain Analysis, Budgetary Control: Flexible Budgets; Performance budgets; Zero-based budgets.

UNIT V QUANTITATIVE TECHNIQUES FOR COST MANAGEMENT 9
Linear Programming, PERT/CPM, Transportation problems, Assignment problems, Learning Curve Theory.

TOTAL: 45 PERIODS

OUTCOMES:

Students will be able to:
CO1 – Understand the costing concepts and their role in decision making
CO2 – Understand the project management concepts and their various aspects in selection
CO3 – Interpret costing concepts with project execution
CO4 – Gain knowledge of costing techniques in service sector and various budgetary control techniques
CO5 - Become familiar with quantitative techniques in cost management
## REFERENCES:
2. Charles T. Horngren and George Foster, Advanced Management Accounting, 1988

## OBJECTIVES:
- Summarize the characteristics of composite materials and effect of reinforcement in composite materials.
- Identify the various reinforcements used in composite materials.
- Compare the manufacturing process of metal matrix composites.
- Understand the manufacturing processes of polymer matrix composites.
- Analyze the strength of composite materials.

## INTRODUCTION
- Definition – Classification and characteristics of Composite materials - Advantages and application of composites - Functional requirements of reinforcement and matrix - Effect of reinforcement (size, shape, distribution, volume fraction) on overall composite performance.

## UNIT II REINFORCEMENTS
- Preparation-layup, curing, properties and applications of glass fibers, carbon fibers, Kevlar fibers and Boron fibers - Properties and applications of whiskers, particle reinforcements - Mechanical Behavior of composites: Rule of mixtures, Inverse rule of mixtures - Isostrain and Isostress conditions.

## UNIT III MANUFACTURING OF METAL MATRIX COMPOSITES

## UNIT IV MANUFACTURING OF POLYMER MATRIX COMPOSITES
UNIT V  STRENGTH
Laminar Failure Criteria-strength ratio, maximum stress criteria, maximum strain criteria, interacting failure criteria, hygrothermal failure. Laminate first play failure-insight strength; Laminate strength-ply discount truncated maximum strain criterion; strength design using caplet plots; stress concentrations.

TOTAL: 45 PERIODS

OUTCOMES:
Students will be able to:
- CO1 - Know the characteristics of composite materials and effect of reinforcement in composite materials.
- CO2 – Know the various reinforcements used in composite materials.
- CO3 – Understand the manufacturing processes of metal matrix composites.
- CO4 – Understand the manufacturing processes of polymer matrix composites.
- CO5 – Analyze the strength of composite materials.

<table>
<thead>
<tr>
<th></th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
<th>PO7</th>
<th>PO8</th>
<th>PO9</th>
<th>PO10</th>
<th>PO11</th>
<th>PO12</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CO2</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>CO3</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CO4</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CO5</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

REFERENCES:

OE5096  WASTE TO ENERGY  L T P C
3 0 0 3

OBJECTIVES:
- Interpret the various types of wastes from which energy can be generated
- Develop knowledge on biomass pyrolysis process and its applications
- Develop knowledge on various types of biomass gasifiers and their operations
- Invent knowledge on biomass combustors and its applications on generating energy
- Summarize the principles of bio-energy systems and their features

UNIT I  INTRODUCTION TO EXTRACTION OF ENERGY FROM WASTE  9
Classification of waste as fuel – Agro based, Forest residue, Industrial waste - MSW – Conversion devices – Incinerators, gasifiers, digestors
UNIT II  BIOMASS PYROLYSIS
Pyrolysis – Types, slow fast – Manufacture of charcoal – Methods - Yields and application – Manufacture of pyrolytic oils and gases, yields and applications.

UNIT III  BIOMASS GASIFICATION

UNIT IV  BIOMASS COMBUSTION
Biomass stoves – Improved chullahs, types, some exotic designs, Fixed bed combustors, Types, inclined grate combustors, Fluidized bed combustors, Design, construction and operation - Operation of all the above biomass combustors.

UNIT V  BIO ENERGY
Properties of biogas (Calorific value and composition), Biogas plant technology and status - Bio energy system - Design and constructional features - Biomass resources and their classification - Biomass conversion processes - Thermo chemical conversion - Direct combustion - biomass gasification - pyrolysis and liquefaction - biochemical conversion - anaerobic digestion - Types of biogas Plants – Applications - Alcohol production from biomass - Bio diesel production -Urban waste to energy conversion - Biomass energy programme in India.

TOTAL: 45 PERIODS

OUTCOMES:
Students will be able to:
CO1 – Understand the various types of wastes from which energy can be generated
CO2 – Gain knowledge on biomass pyrolysis process and its applications
CO3 – Develop knowledge on various types of biomass gasifiers and their operations
CO4 – Gain knowledge on biomass combustors and its applications on generating energy
CO5 – Understand the principles of bio-energy systems and their features

<table>
<thead>
<tr>
<th></th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
<th>PO7</th>
<th>PO8</th>
<th>PO9</th>
<th>PO10</th>
<th>PO11</th>
<th>PO12</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>CO2</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>CO3</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>CO4</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>CO5</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
</tbody>
</table>

REFERENCES:
AUDIT COURSES (AC)

AX5091 ENGLISH FOR RESEARCH PAPER WRITING L T P C 2 0 0 0

OBJECTIVES
- Teach how to improve writing skills and level of readability
- Tell about what to write in each section
- Summarize the skills needed when writing a Title
- Infer the skills needed when writing the Conclusion
- Ensure the quality of paper at very first-time submission

UNIT I INTRODUCTION TO RESEARCH PAPER WRITING 6
Planning and Preparation, Word Order, Breaking up long sentences, Structuring Paragraphs and Sentences, Being Concise and Removing Redundancy, Avoiding Ambiguity and Vagueness

UNIT II PRESENTATION SKILLS 6

UNIT III TITLE WRITING SKILLS 6
Key skills are needed when writing a Title, key skills are needed when writing an Abstract, key skills are needed when writing an Introduction, skills needed when writing a Review of the Literature, Methods, Results, Discussion, Conclusions, The Final Check

UNIT IV RESULT WRITING SKILLS 6
Skills are needed when writing the Methods, skills needed when writing the Results, skills are needed when writing the Discussion, skills are needed when writing the Conclusions

UNIT V VERIFICATION SKILLS 6
Useful phrases, checking Plagiarism, how to ensure paper is as good as it could possibly be the first-time submission

TOTAL: 30 PERIODS

OUTCOMES
CO1 – Understand that how to improve your writing skills and level of readability
CO2 – Learn about what to write in each section
CO3 – Understand the skills needed when writing a Title
CO4 – Understand the skills needed when writing the Conclusion
CO5 – Ensure the good quality of paper at very first-time submission

<table>
<thead>
<tr>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
<th>PO7</th>
<th>PO8</th>
<th>PO9</th>
<th>PO10</th>
<th>PO11</th>
<th>PO12</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>CO2</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>CO3</td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>CO4</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>CO5</td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
</tr>
</tbody>
</table>

REFERENCES
OBJECTIVES

- Summarize basics of disaster
- Explain a critical understanding of key concepts in disaster risk reduction and humanitarian response.
- Illustrate disaster risk reduction and humanitarian response policy and practice from multiple perspectives.
- Describe an understanding of standards of humanitarian response and practical relevance in specific types of disasters and conflict situations.
- Develop the strengths and weaknesses of disaster management approaches.

UNIT I INTRODUCTION
Disaster: Definition, Factors and Significance; Difference between Hazard And Disaster; Natural and Manmade Disasters: Difference, Nature, Types and Magnitude.

UNIT II REPERCUSSIONS OF DISASTERS AND HAZARDS

UNIT III DISASTER PRONE AREAS IN INDIA
Study of Seismic Zones; Areas Prone To Floods and Droughts, Landslides And Avalanches; Areas Prone To Cyclonic and Coastal Hazards with Special Reference To Tsunami; Post-Disaster Diseases and Epidemics.

UNIT IV DISASTER PREPAREDNESS AND MANAGEMENT
Preparedness: Monitoring Of Phenomena Triggering a Disaster or Hazard; Evaluation of Risk: Application of Remote Sensing, Data from Meteorological And Other Agencies, Media Reports: Governmental and Community Preparedness.

UNIT V RISK ASSESSMENT

OUTCOMES

CO1: Ability to summarize basics of disaster
CO2: Ability to explain a critical understanding of key concepts in disaster risk reduction and humanitarian response.
CO3: Ability to illustrate disaster risk reduction and humanitarian response policy and practice from multiple perspectives.
CO4: Ability to describe an understanding of standards of humanitarian response and practical relevance in specific types of disasters and conflict situations.
CO5: Ability to develop the strengths and weaknesses of disaster management approaches.

TOTAL : 30 PERIODS
REFERENCES

AX5093 SANSKRIT FOR TECHNICAL KNOWLEDGE L T P C

OBJECTIVES
• Illustrate the basic sanskrit language.
• Recognize sanskrit, the scientific language in the world.
• Appraise learning of sanskrit to improve brain functioning.
• Relate sanskrit to develop the logic in mathematics, science & other subjects enhancing the memory power.
• Extract huge knowledge from ancient literature.

UNIT I ALPHABETS 6
Alphabets in Sanskrit

UNIT II TENSES AND SENTENCES 6
Past/Present/Future Tense - Simple Sentences

UNIT III ORDER AND ROOTS 6
Order - Introduction of roots

UNIT IV SANSKRIT LITERATURE 6
Technical information about Sanskrit Literature

UNIT V TECHNICAL CONCEPTS OF ENGINEERING 6
Technical concepts of Engineering-Electrical, Mechanical, Architecture, Mathematics

TOTAL: 30 PERIODS

OUTCOMES
• CO1 - Understanding basic Sanskrit language.
• CO2 - Write sentences.
• CO3 - Know the order and roots of Sanskrit.
• CO4 - Know about technical information about Sanskrit literature.
• CO5 - Understand the technical concepts of Engineering.
**REFERENCES**

1. “Abhyaspustakam” – Dr. Vishwas, Samskrita-Bharti Publication, New Delhi
2. “Teach Yourself Sanskrit” Prathama Deeksha-Vempati Kutumbshastri, Rashtriya Sanskrit Sansthanam, New Delhi Publication

**AX5094**

**VALUE EDUCATION**

<table>
<thead>
<tr>
<th>L</th>
<th>T</th>
<th>P</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**OBJECTIVES**
Students will be able to
- Understand value of education and self-development
- Imbibe good values in students
- Let the should know about the importance of character

**UNIT I**

**UNIT II**

**UNIT III**

**UNIT IV**

**TOTAL: 30 PERIODS**

**OUTCOMES**
Students will be able to
- Knowledge of self-development.
- Learn the importance of Human values.
- Developing the overall personality.
Suggested reading

AX5095 CONSTITUTION OF INDIA L T P C 2 0 0 0

OBJECTIVES
Students will be able to:
- Understand the premises informing the twin themes of liberty and freedom from a civil rights perspective.
- To address the growth of Indian opinion regarding modern Indian intellectuals' constitutional
- Role and entitlement to civil and economic rights as well as the emergence nation
  hood in the early years of Indian nationalism.
- To address the role of socialism in India after the commencement of the Bolshevik 
  Revolutionin1917and its impact on the initial drafting of the Indian Constitution.

UNIT I HISTORY OF MAKING OF THE INDIAN CONSTITUTION:
History, Drafting Committee, (Composition & Working)

UNIT II PHILOSOPHY OF THE INDIAN CONSTITUTION:
Preamble, Salient Features

UNIT III CONTOURS OF CONSTITUTIONAL RIGHTS AND DUTIES:
Fundamental Rights, Right to Equality, Right to Freedom, Right against Exploitation, Right to 
Freedom of Religion, Cultural and Educational Rights, Right to Constitutional Remedies, 

UNIT IV ORGANS OF GOVERNANCE:
Parliament, Composition, Qualifications and Disqualifications, Powers and Functions, 
Executive, President, Governor, Council of Ministers, Judiciary, Appointment and Transfer of 
Judges, Qualifications, Powers and Functions.

UNIT V LOCAL ADMINISTRATION:
District’s Administration head: Role and Importance, Municipalities: Introduction, Mayor and 
role of Elected Representative, CEO, Municipal Corporation. Panchayati raj: Introduction, PRI:
Zila Panchayat. Elected officials and their roles, CEO Zila Panchayat: Position and role. Block 
level: Organizational Hierarchy(Different departments), Village level:Role of Elected and 
Appointed officials, Importance of grass root democracy.

UNIT VI ELECTION COMMISSION:
Election Commission: Role and Functioning. Chief Election Commissioner and Election 
Commissioners - Institute and Bodies for the welfare of SC/ST/OBC and women.

TOTAL: 30 PERIODS

OUTCOMES
Students will be able to:
- Discuss the growth of the demand for civil rights in India for the bulk of Indians before 
  the arrival of Gandhi in Indian politics.
- Discuss the intellectual origins of the framework of argument that informed the 
  conceptualization
- of social reforms leading to revolution in India.
• Discuss the circumstances surrounding the foundation of the Congress Socialist Party [CSP] under the leadership of Jawaharlal Nehru and the eventual failure of the proposal of direct elections through adult suffrage in the Indian Constitution.
• Discuss the passage of the Hindu Code Bill of 1956.

SUGGESTED READING
1. The Constitution of India, 1950 (Bare Act), Government Publication.

AX5096 PEDAGOGY STUDIES L T P C
2 0 0 0

OBJECTIVES
Students will be able to:
• Review existing evidence on there view topic to inform programme design and policy
• Making under taken by the DfID, other agencies and researchers.
• Identify critical evidence gaps to guide the development.

UNIT I  INTRODUCTION AND METHODOLOGY:
Aims and rationale, Policy background, Conceptual framework and terminology - Theories of learning, Curriculum, Teacher education - Conceptual framework, Research questions - Overview of methodology and Searching.

UNIT II  THEMATIC OVERVIEW
Pedagogical practices are being used by teachers in formal and informal classrooms in developing countries - Curriculum, Teacher education.

UNIT III  EVIDENCE ON THE EFFECTIVENESS OF PEDAGOGICAL PRACTICES
Methodology for the in depth stage: quality assessment of included studies - How can teacher education (curriculum and practicum) and the school curriculum and guidance materials best support effective pedagogy? - Theory of change - Strength and nature of the body of evidence for effective pedagogical practices - Pedagogic theory and pedagogical approaches - Teachers’ attitudes and beliefs and Pedagogic strategies.

UNIT IV  PROFESSIONAL DEVELOPMENT
Professional development: alignment with classroom practices and follow up support - Peer support - Support from the head teacher and the community - Curriculum and assessment - Barriers to learning: limited resources and large class sizes

UNIT V  RESEARCH GAPS AND FUTURE DIRECTIONS
Research design – Contexts – Pedagogy - Teacher education - Curriculum and assessment - Dissemination and research impact.

TOTAL: 30 PERIODS
OUTCOMES
Students will be able to understand:
- What pedagogical practices are being used by teachers informal and informal classrooms in developing countries?
- What is the evidence on the effectiveness of these pedagogical practices, in what conditions, and with what population of learners?
- How can teacher education (curriculum and practicum) and the school curriculum and guidance materials best support effective pedagogy?

SUGGESTED READING

AX5097 STRESS MANAGEMENT BY YOGA L T P C
2 0 0 0

OBJECTIVES
- To achieve overall health of body and mind
- To overcome stress

UNIT I
Definitions of Eight parts of yoga.(Ashtanga)

UNIT II
Yam and Niyam - Do’s and Don’t’s in life - i) Ahinsa, satya, astheya, bramhacharya and aparigraha, ii) Ahinsa, satya, astheya, bramhacharya and aparigraha.

UNIT III
Asan and Pranayam - Various yog poses and their benefits for mind & body - Regularization of breathing techniques and its effects-Types of pranayam

TOTAL: 30 PERIODS

OUTCOMES
Students will be able to:
- Develop healthy mind in a healthy body thus improving social health also
- Improve efficiency
SUGGESTED READING
1. “Yogic Asanas for Group Tarining-Part-I”: Janardan Swami Yoga Bhyasi Mandal, Nagpur
2. “Rajayoga or conquering the Internal Nature” by Swami Vivekananda, Advaita Ashrama
   (Publication Department), Kolkata

AX5098 PERSONALITY DEVELOPMENT THROUGH LIFE ENLIGHTENMENT SKILLS

OBJECTIVES
- To learn to achieve the highest goal happily
- To become a person with stable mind, pleasing personality and determination
- To awaken wisdom in students

UNIT I
Neetisatakam-holistic development of personality - Verses- 19,20,21,22 (wisdom) - Verses- 29,31,32 (pride & heroism) – Verses- 26,28,63,65 (virtue) - Verses- 52,53,59 (don'ts) - Verses- 71,73,75,78 (do's)

UNIT II
Approach to day to day work and duties - Shrimad Bhagwad Geeta: Chapter 2-Verses 41, 47,48 - Chapter 3-Verses 13, 21, 27, 35 Chapter 6-Verses 5,13,17,23, 35 - Chapter 18-Verses 45, 46, 48.

UNIT III
Statements of basic knowledge - Shrimad Bhagwad Geeta: Chapter2-Verses 56, 62, 68 Chapter 12 -Verses 13, 14, 15, 16,17, 18 - Personality of role model - shrimad bhagwad geeta - Chapter2-Verses 17, Chapter 3-Verses 36,37,42 - Chapter 4-Verses 18, 38,39 Chapter18 – Verses 37,38,63

TOTAL: 30 PERIODS

OUTCOMES
Students will be able to
- Study of Shrimad Bhagwad Geeta will help the student in developing his personality and achieve the highest goal in life
- The person who has studied Geeta will lead the nation and mankind to peace and prosperity
- Study of Neet is hatakanam will help in developing versatile personality of students.

SUGGESTED READING
1. Gopinath, Rashtriya Sanskrit Sansthanam P, Bhartrihi’s Three Satakam, Niti-sringar-vairagya, New Delhi, 2010