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.
M ASTER OF COMPUTER APPLICATIONS

1. PROGRAMME EDUCATIONAL OBJECTIVES (PEOs):

<table>
<thead>
<tr>
<th>PEO#</th>
<th>Programme Educational Objectives</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>To prepare students with breadth of knowledge to comprehend, analyze, design and create computing solutions to real-life problems and to excel in industry/ technical profession.</td>
</tr>
<tr>
<td>2</td>
<td>To provide students with solid foundation in mathematical and computing fundamentals and techniques required to solve technology related problems and to pursue higher studies and research.</td>
</tr>
<tr>
<td>3</td>
<td>To inculcate a professional and ethical attitude in students, to enable them to work towards a broad social context.</td>
</tr>
<tr>
<td>4</td>
<td>To empower students with skills required to work as member and leader in multidisciplinary teams and with continuous learning ability on technology and trends needed for a successful career.</td>
</tr>
</tbody>
</table>

2. PROGRAMME OUTCOMES (POs):

After going through the two years of study, our master's in computer applications Graduates will exhibit ability to:

<table>
<thead>
<tr>
<th>PO#</th>
<th>Programme Outcome</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><strong>Computational Knowledge:</strong> Apply knowledge of computing fundamentals, computing specialisation, mathematics, and domain knowledge appropriate for the computing specialisation to the abstraction and conceptualisation of computing models from defined problems and requirements.</td>
</tr>
<tr>
<td>2</td>
<td><strong>Problem Analysis:</strong> Identify, formulate, research literature and solve complex computing problems reaching substantiated conclusions using fundamental principles of mathematics, computing sciences, and relevant domain disciplines.</td>
</tr>
<tr>
<td>3</td>
<td><strong>Design /Development of Solutions:</strong> Design and evaluate solutions for complex computing problems, and design and evaluate systems, components, or processes that meet specified needs with appropriate consideration for public health and safety, cultural, societal, and environmental considerations.</td>
</tr>
<tr>
<td>4</td>
<td><strong>Conduct investigations of complex Computing problems:</strong> Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.</td>
</tr>
<tr>
<td>5</td>
<td><strong>Modern Tool Usage:</strong> Create, select, adapt and apply appropriate techniques, resources, and modern computing tools to complex computing activities, with an understanding of the limitations.</td>
</tr>
<tr>
<td></td>
<td><strong>Professional Ethics:</strong> Understand and commit to professional ethics and cyber regulations, responsibilities, and norms of professional computing practices.</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>7.</td>
<td><strong>Life-long Learning:</strong> Recognise the need, and have the ability, to engage in independent learning for continual development as a computing professional.</td>
</tr>
<tr>
<td>8.</td>
<td><strong>Project management and finance:</strong> Demonstrate knowledge and understanding of computing and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.</td>
</tr>
<tr>
<td>9.</td>
<td><strong>Communication Efficacy:</strong> Communicate effectively with the computing community, and with society at large, about complex computing activities by being able to comprehend and write effective reports, design documentation, make effective presentations, and give and understand clear instructions.</td>
</tr>
<tr>
<td>10.</td>
<td><strong>Societal and Environmental Concern:</strong> Understand and assess societal, environmental, health, safety, legal, and cultural issues within local and global contexts, and the consequential responsibilities relevant to professional computing practices.</td>
</tr>
<tr>
<td>11.</td>
<td><strong>Individual and Team Work:</strong> Function effectively as an individual and as a member or leader in diverse teams and in multidisciplinary environments.</td>
</tr>
<tr>
<td>12.</td>
<td><strong>Innovation and Entrepreneurship:</strong> Identify a timely opportunity and using innovation to pursue that opportunity to create value and wealth for the betterment of the individual and society at large.</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>
<th>PO7</th>
<th>PO8</th>
<th>PO9</th>
<th>PO10</th>
<th>PO11</th>
<th>PO12</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</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>II</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>III</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>IV</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>

### Mapping of Course Outcome and Programme Outcome

<table>
<thead>
<tr>
<th>Year</th>
<th>COURSE TITLE</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></td>
<td>Linear Algebra, Probability and Statistics</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></td>
<td>Data Structures &amp; Algorithms</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>YEAR 1</td>
<td>Python Programming</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></td>
<td>Network Programming and Device Management</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></td>
<td>Content Technologies</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></td>
<td>Research Methodology and IPR</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></td>
<td>Audit Course I</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></td>
<td>Programming in Python Laboratory</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></td>
<td>Data structures &amp; Algorithms Laboratory</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>SEM 1</td>
<td>Cloud Computing Techniques</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></td>
<td>Data Analytics</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></td>
<td>Advanced Java Programming</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></td>
<td>Full Stack Software Development</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></td>
<td>Full Stack Laboratory</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></td>
<td>Professional Elective I</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></td>
<td>Professional Elective II</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></td>
<td>Audit Course II</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></td>
<td>Mobile Application Development Laboratory</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>Year</td>
<td>COURSE TITLE</td>
<td>PO 1</td>
<td>PO 2</td>
<td>PO 3</td>
<td>PO 4</td>
<td>PO 5</td>
<td>PO 6</td>
<td>PO 7</td>
<td>PO 8</td>
<td>PO 9</td>
<td>PO 10</td>
<td>PO 11</td>
<td>PO 12</td>
</tr>
<tr>
<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>-------</td>
<td>-------</td>
</tr>
<tr>
<td></td>
<td>Artificial Intelligence &amp; Machine Learning</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></td>
<td>Internet of Things</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></td>
<td>Cyber Security</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>SEM 3</td>
<td>Professional Elective III</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></td>
<td>Professional Elective IV</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></td>
<td>Professional Elective V</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></td>
<td>Open Elective</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></td>
<td>Machine Learning Techniques Laboratory</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>SEM 4</td>
<td>Project Work</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>
### SEMESTER I

<table>
<thead>
<tr>
<th>S.N.O.</th>
<th>COURSE CODE</th>
<th>COURSE TITLE</th>
<th>CATEG-ORY</th>
<th>PERIODS PER WEEK</th>
<th>TOTAL CONTACT PERIODS</th>
<th>CREDITS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td><strong>THEORY</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.</td>
<td>MA5104</td>
<td>Linear Algebra, Probability and Statistics</td>
<td>FC</td>
<td>3 1 0</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>2.</td>
<td>CA5104</td>
<td>Data Structures &amp; Algorithms</td>
<td>PCC</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>3.</td>
<td>CA5105</td>
<td>Python Programming</td>
<td>PCC</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>4.</td>
<td>CA5106</td>
<td>Content Technologies</td>
<td>PCC</td>
<td>3 0 2</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>5.</td>
<td>CA5107</td>
<td>Network Programming and Device Management</td>
<td>PCC</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>6.</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>7.</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><strong>PRACTICALS</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8.</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>9.</td>
<td>CA5113</td>
<td>Data Structures &amp; Algorithms 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>19 1 10</td>
<td>30</td>
<td>23</td>
</tr>
</tbody>
</table>

*Audit course is optional

### SEMESTER II

<table>
<thead>
<tr>
<th>S.N.O.</th>
<th>COURSE CODE</th>
<th>COURSE TITLE</th>
<th>CATEG-ORY</th>
<th>PERIODS PER WEEK</th>
<th>TOTAL CONTACT PERIODS</th>
<th>CREDITS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td><strong>THEORY</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.</td>
<td>CA5206</td>
<td>Data Analytics</td>
<td>PCC</td>
<td>3 0 2</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>2.</td>
<td>CA5207</td>
<td>Advanced Java Programming</td>
<td>PCC</td>
<td>3 0 2</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>3.</td>
<td>CA5208</td>
<td>Full Stack Software Development</td>
<td>PCC</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>4.</td>
<td>CA5209</td>
<td>Cloud Computing Techniques</td>
<td>PCC</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>5.</td>
<td></td>
<td>Professional Elective I</td>
<td>PEC</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>6.</td>
<td></td>
<td>Professional Elective II</td>
<td>PEC</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>7.</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><strong>PRACTICALS</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8.</td>
<td>CA5213</td>
<td>Full Stack Laboratory</td>
<td>EEC</td>
<td>0 0 4</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>9.</td>
<td>CA5511</td>
<td>Mobile Application Development Laboratory</td>
<td>EEC</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>20 0 12</td>
<td>32</td>
<td>24</td>
</tr>
</tbody>
</table>

*Audit course is optional
# Semester III

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Course Code</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>CA5304</td>
<td>Artificial Intelligence &amp; Machine Learning</td>
<td>PCC</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>2.</td>
<td>CA5305</td>
<td>Internet of Things</td>
<td>PCC</td>
<td>3 0 2</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>3.</td>
<td>CA5306</td>
<td>Cyber Security</td>
<td>PCC</td>
<td>3 0 2</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>4.</td>
<td></td>
<td>Professional Elective III</td>
<td>PEC</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>5.</td>
<td></td>
<td>Professional Elective IV</td>
<td>PEC</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>6.</td>
<td></td>
<td>Professional Elective V</td>
<td>PEC</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>7.</td>
<td></td>
<td>Open Elective</td>
<td>OEC</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Practical</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8.</td>
<td>CA5314</td>
<td>Machine Learning Techniques Laboratory</td>
<td>EEC</td>
<td>0 0 4</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TOTAL</td>
<td></td>
<td>21 0 8</td>
<td>29</td>
<td>25</td>
</tr>
</tbody>
</table>

# Semester IV

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Course Code</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>Practical</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.</td>
<td>CA 5414</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>TOTAL</td>
<td></td>
<td>3 0 24</td>
<td>27</td>
<td>12</td>
</tr>
</tbody>
</table>

**Total Credits: 84**

# List of Foundation Courses (FC)

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Course Code</th>
<th>Course Title</th>
<th>Periods Per Week</th>
<th>Total Contact Periods</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>MA5104</td>
<td>Linear Algebra, Probability and Statistics</td>
<td>3 1 0</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TOTAL</td>
<td>3 1 0</td>
<td>4</td>
<td>4</td>
</tr>
</tbody>
</table>
### LIST OF PROFESSIONAL CORE COURSES (PCC)

<table>
<thead>
<tr>
<th>S. NO.</th>
<th>COURSE CODE</th>
<th>COURSE TITLE</th>
<th>PERIODS PER WEEK</th>
<th>TOTAL CONTACT PERIODS</th>
<th>CREDITS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>CA5104</td>
<td>Data Structures &amp; Algorithms</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>2.</td>
<td>CA5105</td>
<td>Python Programming</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>3.</td>
<td>CA5107</td>
<td>Network Programming and Device Management</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>4.</td>
<td>CA5106</td>
<td>Content Technologies</td>
<td>3 0 2</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>5.</td>
<td>CA5111</td>
<td>Programming in Python Laboratory</td>
<td>0 0 4</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>6.</td>
<td>CA5113</td>
<td>Data structures &amp; Algorithms Laboratory</td>
<td>0 0 4</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>7.</td>
<td>CA5209</td>
<td>Cloud Computing Techniques</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>8.</td>
<td>CA5206</td>
<td>Data Analytics</td>
<td>3 0 2</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>9.</td>
<td>CA5207</td>
<td>Advanced Java Programming</td>
<td>3 0 2</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>10.</td>
<td>CA5208</td>
<td>Full Stack Software Development</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>11.</td>
<td>CA5304</td>
<td>Artificial Intelligence &amp; Machine Learning</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>12.</td>
<td>CA5305</td>
<td>Internet of Things</td>
<td>3 0 2</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>13.</td>
<td>CA5306</td>
<td>Cyber Security</td>
<td>3 0 2</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>TOTAL</td>
<td></td>
<td>33 0 18</td>
<td>51</td>
<td>42</td>
</tr>
</tbody>
</table>

### LIST OF PROFESSIONAL ELECTIVES (PEC)

<table>
<thead>
<tr>
<th>S. NO.</th>
<th>COURSE CODE</th>
<th>COURSE TITLE</th>
<th>PERIODS PER WEEK</th>
<th>TOTAL CONTACT PERIODS</th>
<th>CREDITS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>CA5001</td>
<td>Blockchain Technologies</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>2.</td>
<td>CA5002</td>
<td>Ethical Hacking</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>3.</td>
<td>CA5003</td>
<td>Big Data with R</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>4.</td>
<td>CA5007</td>
<td>E-Learning Techniques</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>5.</td>
<td>CA5008</td>
<td>Software Testing</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>6.</td>
<td>CA5009</td>
<td>Deep Learning Techniques and Applications</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>7.</td>
<td>CA5010</td>
<td>Game Programming Techniques</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>8.</td>
<td>CA5011</td>
<td>Multimedia Technologies</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>9.</td>
<td>CA5012</td>
<td>Data Visualization Techniques</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>10.</td>
<td>CA5014</td>
<td>C# and .NET Programming</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>11.</td>
<td>CA5015</td>
<td>Service Oriented Architectures</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>12.</td>
<td>CA5016</td>
<td>Software Project Management</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>13.</td>
<td>CA5017</td>
<td>Mixed Reality</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>14.</td>
<td>CA5018</td>
<td>Digital Image Processing and Applications</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>S. NO.</td>
<td>COURSE CODE</td>
<td>COURSE TITLE</td>
<td>PERIODS PER WEEK</td>
<td>TOTAL CONTACT PERIODS</td>
<td>CREDITS</td>
</tr>
<tr>
<td>-------</td>
<td>-------------</td>
<td>--------------------------------------------------</td>
<td>------------------</td>
<td>-----------------------</td>
<td>---------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>L</td>
<td>T</td>
<td>P</td>
</tr>
<tr>
<td>15.</td>
<td>CA5019</td>
<td>Text Mining Techniques</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16.</td>
<td>CA5020</td>
<td>Data Warehousing and Data Mining Techniques</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>17.</td>
<td>CA5021</td>
<td>Software Quality Assurance</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>18.</td>
<td>CA5022</td>
<td>Introduction to Social Network Analysis</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>19.</td>
<td>CA5028</td>
<td>Wireless Sensor Networks &amp; Protocols</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>20.</td>
<td>CA5032</td>
<td>Semantic Web and Applications</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>21.</td>
<td>CA5033</td>
<td>Soft Computing</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>22.</td>
<td>CA5034</td>
<td>Cognitive Computing</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>23.</td>
<td>CA5035</td>
<td>Middleware Technologies</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>24.</td>
<td>CA5036</td>
<td>UI &amp; UX</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>25.</td>
<td>CA5037</td>
<td>Robotic Process Automation</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>26.</td>
<td>CA5038</td>
<td>Software Design Principles &amp; Architecture Patterns</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>27.</td>
<td>CA5039</td>
<td>Autonomous Ground Vehicle and Unmanned Aerial Vehicle Systems</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>28.</td>
<td>CA5040</td>
<td>Financial Technologies</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>29.</td>
<td>CA5041</td>
<td>Linux Administration</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>30.</td>
<td>CA5042</td>
<td>Voice Technology</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>31.</td>
<td>CA5043</td>
<td>Business Domains and Verticals</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>32.</td>
<td>CA5044</td>
<td>Digital Forensics</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>TOTAL</strong></td>
<td>96</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**LIST OF RESEARCH METHODOLOGY AND IPR COURSES (RMC)**

<table>
<thead>
<tr>
<th>S. NO.</th>
<th>COURSE CODE</th>
<th>COURSE TITLE</th>
<th>PERIODS PER WEEK</th>
<th>TOTAL CONTACT PERIODS</th>
<th>CREDITS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>L</td>
<td>T</td>
<td>P</td>
</tr>
<tr>
<td>1.</td>
<td>RM5151</td>
<td>Research Methodology and IPR</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>TOTAL</strong></td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
### LIST OF OPEN ELECTIVE COURSES (OEC)
*(out of 6 courses one course must be selected)*

<table>
<thead>
<tr>
<th>S. NO</th>
<th>COURSE CODE</th>
<th>COURSE TITLE</th>
<th>PERIODS PER WEEK</th>
<th>TOTAL CONTACT PERIODS</th>
<th>CREDITS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>OE5091</td>
<td>Business Data Analytics</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>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>3.</td>
<td>OE5093</td>
<td>Operations Research</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>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>5.</td>
<td>OE5095</td>
<td>Composite Materials</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>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>TOTAL</strong></td>
<td><strong>18 0 0</strong></td>
<td><strong>18 0</strong></td>
<td><strong>18</strong></td>
</tr>
</tbody>
</table>

### LIST OF AUDIT COURSES (AC)
Registration for any of these courses is optional to students

<table>
<thead>
<tr>
<th>S. NO</th>
<th>COURSE CODE</th>
<th>COURSE TITLE</th>
<th>PERIODS PER WEEK</th>
<th>TOTAL CONTACT PERIODS</th>
<th>CREDITS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>AX5091</td>
<td>English for Research Paper Writing</td>
<td>2 0 0</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>2.</td>
<td>AX5092</td>
<td>Disaster Management</td>
<td>2 0 0</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>3.</td>
<td>AX5093</td>
<td>Sanskrit for Technical Knowledge</td>
<td>2 0 0</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>4.</td>
<td>AX5094</td>
<td>Value Education</td>
<td>2 0 0</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>5.</td>
<td>AX5095</td>
<td>Constitution of India</td>
<td>2 0 0</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>6.</td>
<td>AX5096</td>
<td>Pedagogy Studies</td>
<td>2 0 0</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>7.</td>
<td>AX5097</td>
<td>Stress Management by Yoga</td>
<td>2 0 0</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>8.</td>
<td>AX5098</td>
<td>Personality Development Through Life Enlightenment Skills</td>
<td>2 0 0</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>9.</td>
<td>AX5099</td>
<td>Unnat Bharat Abhiyan</td>
<td>2 0 0</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>TOTAL</strong></td>
<td><strong>18 0 0</strong></td>
<td><strong>18 0</strong></td>
<td><strong>0</strong></td>
</tr>
</tbody>
</table>

### LIST OF EMPLOYABLITY ENHANCEMENT COURSES (EEC)

<table>
<thead>
<tr>
<th>S.No</th>
<th>COURSE CODE</th>
<th>COURSE TITLE</th>
<th>PERIODS PER WEEK</th>
<th>TOTAL CONTACT PERIODS</th>
<th>CREDITS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>CA5213</td>
<td>Full Stack Laboratory</td>
<td>0 0 4</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>2.</td>
<td>CA5511</td>
<td>Mobile Application Development Laboratory</td>
<td>0 0 4</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>3.</td>
<td>CA5314</td>
<td>Machine Learning Techniques Laboratory</td>
<td>0 0 4</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>4.</td>
<td>CA5414</td>
<td>Project Work</td>
<td>0 0 2</td>
<td>4</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>TOTAL</strong></td>
<td><strong>0 0 36</strong></td>
<td><strong>36 18</strong></td>
<td><strong>18</strong></td>
</tr>
</tbody>
</table>
## LIST OF BRIDGE COURSES (BC)

<table>
<thead>
<tr>
<th>S. NO</th>
<th>COURSE CODE</th>
<th>COURSE TITLE</th>
<th>PERIODS PER WEEK</th>
<th>TOTAL CONTACT PERIODS</th>
<th>CREDITS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>L</td>
<td>T</td>
<td>P</td>
</tr>
<tr>
<td>Semester I</td>
<td></td>
<td></td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1.</td>
<td>MA5105</td>
<td>Mathematical Foundations of Computer Science</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2.</td>
<td>BX5001</td>
<td>Fundamentals of Computing</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3.</td>
<td>BX5004</td>
<td>Basic Data Structures</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4.</td>
<td>BX5006</td>
<td>Software Engineering</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5.</td>
<td>BX5008</td>
<td>C Programming and Data Structures Laboratory</td>
<td>0</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>Semester II</td>
<td></td>
<td></td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>6.</td>
<td>BX5002</td>
<td>Digital Logic &amp; Computer Organization</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>7.</td>
<td>BX5003</td>
<td>Operating systems</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>8.</td>
<td>BX5005</td>
<td>Database Management Systems</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>9.</td>
<td>BX5007</td>
<td>Java programming</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>10.</td>
<td>BX5009</td>
<td>Database Management Systems Laboratory</td>
<td>0</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>11.</td>
<td>BX5010</td>
<td>Java Programming Laboratory</td>
<td>0</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>TOTAL</td>
<td></td>
<td></td>
<td>24</td>
<td>0</td>
<td>12</td>
</tr>
</tbody>
</table>

### SEMESTER-WISE CREDIT DISTRIBUTION

<table>
<thead>
<tr>
<th>S. NO</th>
<th>CATEGORY</th>
<th>CREDITS PER SEMESTER</th>
<th>CREDITS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>I</td>
<td>II</td>
<td>III</td>
</tr>
<tr>
<td>1.</td>
<td>Foundation Courses (FC)</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>2.</td>
<td>Professional Core Courses (PCC)</td>
<td>17</td>
<td>14</td>
</tr>
<tr>
<td>3.</td>
<td>Professional Elective Courses (PEC)</td>
<td>0</td>
<td>6</td>
</tr>
<tr>
<td>4.</td>
<td>Open Elective Courses (OEC)</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5.</td>
<td>Research Methodology and IPR Courses (RMC)</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>6.</td>
<td>Employability Enhancement Courses (EEC)</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>7.</td>
<td>Audit Courses (AC)</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>TOTAL</td>
<td></td>
<td>23</td>
<td>24</td>
</tr>
</tbody>
</table>
OBJECTIVES:
- To find the basis and dimension of vector space
- To obtain the matrix of linear transformation and its eigenvalues and eigenvectors
- To provide foundation on Applied Probability
- To use various statistical techniques in Application problems
- To introduce the concept of Design of Experiments for data analysis.

UNIT I VECTOR SPACES
Real and Complex fields - Vector spaces over Real and Complex fields - Sub space - Linear space - Linear independence and dependence - Basis and dimension.

UNIT II LINEAR TRANSFORMATION
Linear transformation - Rank space and null space - Rank and nullity - Dimension theorem - Matrix representation of linear transformation - Eigenvalues and Eigenvectors of linear transformation.

UNIT III PROBABILITY AND RANDOM VARIABLES

UNIT IV TESTING OF HYPOTHESIS
Sampling distributions - Tests based on small and large samples - Normal, Student’s t, Chi-square and F distributions for testing of mean, variance and proportion and testing of difference of means variances and proportions - Tests for independence of attributes and goodness of fit.

UNIT V DESIGN OF EXPERIMENTS
Analysis of variance - Completely randomized design - Random block design (One-way and Two-way classifications) - Latin square design -2² Factorial design.

TOTAL: 60 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
1. Test the consistency and solve system of linear equations.
2. Find the basis and dimension of vector space.
3. Apply the Probability axioms as well as rules and the distribution of discrete and continuous also the random variable ideas in solving real world problems.
4. Use statistical techniques in testing hypothesis on data analysis.
5. Use the appropriate statistical technique of design of experiments in data analysis.

REFERENCES:
2. Faires J.D. and Burden R., Numerical Methods, Brooks/Cole (Thomson Publications),
New Delhi, 2002.

CA5104 DATA STRUCTURES & ALGORITHMS  L  T  P  C
3 0 0 3

OBJECTIVES:

- To learn the concepts of linear data structures and its applications.
- To understand the concepts of non-linear data structures like trees and graphs.
- To learn the usage of sorting techniques.
- To familiarize the concepts of hashing.
- To understand about algorithm analysis and design techniques.

UNIT I LINEAR DATA STRUCTURES


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 II 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 III  HIERARCHICAL DATA STRUCTURES & HASHING  9

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

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

UNIT IV  SORTING, TREES AND GRAPHS  12

Suggested Activities:
- Flipped classroom different sorting techniques such as Bubble Sort, Selection Sort etc.
- External learning - Search algorithms, priority queues, external sorting, replacement selection technique.
- 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.
- Practical - Solving a given problem using efficient search technique.
- 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:
- Quizzes on basics of hashing and sorting.
- Assignments on creation and manipulation of hash table, priority queues.
- Demonstration of the practical implementations.
- Assignments on problem solving on 2-3 trees and M-ary trees.
- Quizzes on binary trees and tree traversals.
- Assignments on applications of graphs and DFS.
- Quizzes on graph operations.

UNIT V  ALGORITHM DESIGN TECHNIQUES  9
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, checkerboard and job selection.

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

OUTCOMES:
On completion of the course, the students will be able to:
1. Use abstract data types including stacks, queues and lists for any application.
2. Design and implement tree data structures.
3. Analyze and implement hashing techniques that solve in linear time.
4. Apply sorting algorithms for a given problem.
5. Design algorithms using graph structures to solve real life problems.
6. Choose appropriate data structure and implement a given application.

REFERENCES:

CA5105 PYTHON PROGRAMMING

OBJECTIVES:
- To know the basics of Python programming.
- 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 work with input/output files, and scripts.

UNIT I VARIABLES, OPERATORS AND CONDITIONALS

Suggested Activities:
- Developing algorithms for basic mathematical expressions using arithmetic operations.
- Installing Python.
- Simple programs on print statements, arithmetic operations.
- Simple Python program implementation using Operators, Conditionals, Iterative Constructs and Functions.

TOTAL: 45 PERIODS
Suggested Evaluation Methods:
- Tutorials on Python programs.

UNIT II  LOOPS AND FUNCTIONS  9

Suggested Activities:
- 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  10

Suggested Activities:
- Practical - Implementing Python program using lists, tuples, sets for the following scenario:
  o Simple Sorting techniques
  o Student Examination Report
  o 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  9

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, EXCEPTION HANDLING AND SYSTEM LEVEL COMMANDS  8

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.
- Automate system tasks using scripts
- Case Studies.

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
3. Structure simple Python programs and functions for solving problems.
5. Read and write data from/to files in Python programs.
6. Automate operating system level tasks using Scripts

REFERENCES:

OBJECTIVES:
- To learn the fundamentals of document databases technologies
- To study the working principles of graph and column databases.
- To have an introductory knowledge about the distributed database patterns.
- To understand the basics of digital asset management and its importance.
- To learn DAM metrics and strategies.

UNIT I DOCUMENT DATABASES

Suggested Activities:
1. Creation of XML Schema using software tools and validating XML documents
2. Creation of JSON documents and realizing and using with AJAX
3. Database manipulation using MongoDB (Creation / Insertion / Updation / Deletion)

Suggested Evaluation Methods:
- Program based evaluation
- Assignments using appropriate technology / tool
- Simple projects using these technologies

UNIT II  GRAPH AND COLUMN DATABASES
9
RDBMS Patterns for Graphs - RDF and SPARQL - Property Graphs and Neo4j - Graph Database Internals - Column Databases - Data Warehousing Schemas - Limitations - The Columnar Alternative- Columnar Compression - Columnar Write Penalty - Column Database Architectures - Projections - Columnar Technology in Other Databases - SSD and In-Memory Databases - Solid State Disk - SSD-Enabled Databases - In-Memory Databases - Redis - Oracle 12c “in-Memory Database” - Spark Architecture.

Suggested Activities:
4. Creation of RDF and OWL based semantics documents
5. Creation of column databases with various technologies
6. Exploring In-memory Databases supported in various technologies

Suggested Evaluation Methods:
- Program based evaluation
- Assignments using appropriate technology / tool
- Simple projects using these technologies

UNIT III DISTRIBUTED DATABASE PATTERNS
9
Distributed Relational Databases - Replication - Nonrelational Distributed Databases - MongoDB Sharding and Replication - Sharding Mechanisms - Cluster Balancing - Replication - HBase - Tables, Regions, and Region Servers - Caching and Data Locality - Cassandra - Consistency Models - ACID and MVCC - Consistency in MongoDB.

Suggested Activities:
- Creation of nonrelational distributed databases
- Practical solutions based on HBase
- Exploring consistency models in MongoDB

Suggested Evaluation Methods:
- Program based evaluation
- Assignments using appropriate technology / tool
- Simple projects using these technologies

UNIT IV DIGITAL ASSETS MANAGEMENT
9

Suggested Activities:
- Realizing DAM server environment
- Creation and accessing assets
- Search strategies for asset management

Suggested Evaluation Methods:
- Program based evaluation
Assignments using appropriate technology / tool
Simple projects using these technologies

UNIT V  DAM METRICS AND STRATEGIES  9
Describing and Searching Mass Sets - Content Audit and Determining Metrics - Building Successful DAM workflow - Digital Preservation and Content Migration Strategies - Brand and Rights Management - DIGITAL ASSET API - Digital Asset Metadata API.

Suggested Activities:
- Building DAM workflow
- Creation and Digital Asset API
- Brand and Rights Management

Suggested Evaluation Methods:
- Assignments
- Comparison of DAM strategies
- Group discussion

EXERCISES:  30
1. Using XML and allied technologies create a database.
2. Using schema validate the XML document.
3. Creation of JSON document and implementing data manipulation.
4. Creation of RDF document and using it with SPARQL.
5. Exploring the significance of SSB and in memory databases.
6. Working with MongoDB for data manipulation.
7. Sharding and Replications using MongoDB environment.
8. Study of HBase.
10. Understanding consistency Model in MongoDB.
12. DA on technologies and search strategies.
14. Using digital Asset API.
15. Study of Brand and Right Management.

OUTCOMES:
On completion of the course, the student will be able to:
1. Able to handle data through XML format and use associated technologies.
2. Use the JSON document databases and manipulation of data.
3. Handle the graph and column databases.
4. Handle large volume of data with distributed database patterns.
5. Get an insight about digital asset management and using search strategies.
6. Use the digital asset API to handle various digital assets.

REFERENCES:
3. Digital Asset Management (DAM) API Guide.
OBJECTIVES:
- To learn the fundamentals of networking, network security and network programming.
- To explore the end to end issues in data communication and possible secure solutions.
- To study about routing algorithms and protocols, IP and related protocols
- To understand medium access control and LANs
- To learn about networking devices and the techniques required to monitor and manage them

UNIT I  ARCHITECTURE AND APPLICATION 9
Data networks – Network Architecture - ISO/OSI and TCP/IP reference models —HTTP and HTTPS, FTP, E-mail and DNS

Suggested Activities:
- Accessing HTTP and SMTP server through Telnet
- External learning - HTTP/DNS format using a tool like Wireshark.
- External learning - POP3 and IMAP protocols of email application

Suggested Evaluation Methods:
- Quiz on Wireshark.
- Quiz on POP3 and IMAP.
- Assignment problems different Application protocols

UNIT II  SOCKET PROGRAMMING 9
System calls and socket programming, Elementary TCP and UDP socket - Developing client/server applications –Socket Options - Advanced Socket IP options for IPv6 server and client’s interoperability - Raw Sockets.

Suggested Activities:
- Assignments on client/server programming
- Assignment on socket options for specific scenarios.
- Practical - Implementation of DNS operation with gethostbyname, gethostbyaddr, getservbyname and getservbyport functions

Suggested Evaluation Methods:
- Quizzes on System calls
- Quiz on raw sockets
- Testing for the respective socket option's role in the scenario chosen.

UNIT III  SECURE COMMUNICATION 9

Suggested Activities:
- Exploring TCP Vegas and TCP Reno
- Flipped classroom on basic encryption and decryption techniques

Suggested Evaluation Methods:
- Quiz on IPSec
- Quizzes on SSL and TLS.

UNIT IV  L2 AND L3 PROTOCOLS AND DEVICES 9
Addressing, VLSM, CIDR - IPv6 – Network devices – Hubs, Bridges, Switches, Routers, L3 Switches.

**Suggested Activities:**
- Discussion on deterministic and non-deterministic MAC protocols
- Flipped classroom on basic of IPv4
- Flip classroom: Configuring routers and congestion control in network layer

**Suggested Evaluation Methods:**
- Analyzing CSMA/CA for various scenarios in terms of hidden and exposed station problems
- Problem solving in VLSM and CIDR
- Assignment on migration from IPv4 to IPv6

**UNIT V DEVICES, MONITORING AND MANAGEMENT**

Edge and Core Networks – Introduction to SDN- data plane- control plane Honeypots – Firewalls – Network monitoring - IDS – Network Management System – SNMP and its variants

**Suggested Activities:**
- Extra learning: Datadog Network Performance Monitoring (FREE TRIAL)
- Flipped classroom: Configuring routers and congestion control in network layer
- Tutorial on SNORT

**Suggested Evaluation Methods:**
- Designing networks using various networking devices as per the given specifications
- Assignment on widely used IDSs and network management tools

**TOTAL: 45 PERIODS**

**OUTCOMES:**
On completion of the course, the students will be able to:
1. Design and implement simple client/server programs using socket options.
2. Design and implement simple cryptosystems
3. To configure various transport layer level parameters in setting up a network
4. To configure various network level parameters in setting up a network
5. Experiment with various tools to manage a network
6. Design a network as per the given specifications, monitor and manage the networks

**REFERENCES**
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  6
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  6
Effective literature studies approaches, analysis, plagiarism, and research ethics.

UNIT III  TECHNICALWRITING /PRESENTATION  6
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)  6

UNIT V  INTELLECTUAL PROPERTY RIGHTS (IPR)  6

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.

REFERENCES:
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.

TOTAL: 60 PERIODS

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 Stack ADT, Queue ADT, and List ADT.
2. Implementation of Binary Search tree and its operations.
4. Implementation of Hashing techniques such as quadratic probing and separate chaining.
5. Implementation of basic 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 algorithm.
9. Implementation of iterative and recursive algorithms with its complexity analysis.
10. Implementation of merge sort algorithm analysis using divide and conquer approach.
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.

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

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.

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.
5. Write a program to construct a Bayesian network considering medical data. Use this model to demonstrate the diagnosis of heart patients using standard Heart Disease Data Set. You can use Java/Python ML library classes/API
6. Implement univariate, bivariate, and multivariate analysis using R Studio.
2. Given a data set, explore the features using data analysis in R.
3. Solve numerical problems on Eigen Value, Eigen Vector, etc. to understand the working principles of mining techniques.
4. Implement an MR program that processes a weather dataset R
5. Implement the following using Hadoop, Map Reduce, HDFS, Hive:
   a. Perform setting up and installing Hadoop in its two operating modes: pseudo-distributed and fully-distributed.
   b. Implement the following file management tasks in Hadoop: adding files and directories, retrieving files and deleting files.
   c. (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.
   d. 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.
   e. Install, deploy & configure Apache Spark cluster. Run Apache Spark applications using Scala.
   f. Install and run Hive then use Hive to create, alter, and drop databases, tables, views, functions, and indexes.

TOTAL: 75 PERIODS

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 large data.
5. Implement suitable data analysis for stream data.
6. Write efficient MapReduce programs for small problem-solving methods.

REFERENCES:
CA5207  ADVANCED JAVA PROGRAMMING

L T P C
3 0 2 4

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 datagrams.

Suggested Evaluation Methods:

- Quiz on generics and networking.
- Tutorial assignments on advanced Java features.

UNIT III  JDBC AND WEB APPLICATION DEVELOPMENT

Accessing Database with JDBC – Basics – Manipulating Databases with JDBC – Overview of JSP – JSP Objects - Running JSP with Database Connectivity – Session Tracking – Java Server Faces

**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 JSP and JSF.

**UNIT IV DISTRIBUTED OBJECTS**

**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**

**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.

**EXERCISES:**
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 JSP and JSF.
8. Session Management and Implementation of Cookies using JSF.
9. Development of SOAP and REST based web services.
2. Web application development using Struts framework & Spring framework.

**TOTAL: 75 PERIODS**

**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:**

CA5208 FULL STACK SOFTWARE DEVELOPMENT L T P C 3 0 0 3

**OBJECTIVES:**
- To understand the basics of JavaScript and importance of MERN stack
- To understand the role of React in designing front-end components
- To understand the design issues in the development of backend components using Node.js and Express
- To understand the significance of using MongoDB as a database system
- To understand the advanced features of full stack development

**UNIT I JAVASCRIPT AND BASICS OF MERN STACK**

**Suggested Activities:**
- Modern JavaScript features-based programming
- Setting up MERN environment
- Simple programs in MERN environment
Suggested Evaluation Methods:
- Programming exercise on JavaScript basic and advanced features
- Case study based simple projects

UNIT II REACT
React Introduction - React ES6 - React Render HTML - React JSX - Components - React Classes - Composing Components - Passing Data - Dynamic Composition - React state - setting State - Async State Initialization - Event Handling Communicating from Child to Parent - Stateless Components - Designing components - React Forms - React CSS - React SaaS

Suggested Activities:
- REACT based programming
- Exploring stateless components
- Designing components with React CSS and SaaS

Suggested Evaluation Methods:
- Programming exercise on REACT based component development
- Simple projects for specific use cases

Unit III NODE.JS AND EXPRESS

Suggested Activities:
- Node and Express based web development Handling of various APIs associated with Node.js

Suggested Evaluation Methods:
- Programming exercise on Node.js based development
- Simple projects for specific use cases

Unit IV MONGODB

Suggested Activities:
- Setting up MongoDB and handling data manipulation
- Querying the MongoDB databases
- Exploring Aggregation, Replication, Sharding and other features in MongoDB

Suggested Evaluation Methods:
- Data manipulation exercises (CRUD) / assignments using MongoDB
- Simple projects for specific use cases

Unit V ADVANCED FEATURES
Suggested Activities:
- Experiments on React Router and various APIs
- Server rendering and handling states
- Session handling

Suggested Evaluation Methods:
- Assignments on using various APIs
- Simple projects for specific use cases

TOTAL: 45 PERIODS

REFERENCES

Web Reference
- http://tutorialsteacher.com
- https://reactjs.org/
- https://nodejs.org
- www.Expressjs.com
- www.mongoddb.com

CA5209 CLOUD COMPUTING TECHNIQUES  L T P C
3 0 0 3

OBJECTIVES:
- To learn 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 9

Suggested Activities:
- Practical - Implement RPC.
- Create and distribute a Torrent file to share a file in LAN environment.
- Create a simple web service using Python Flask/Java/any language [Web Service: Client-server model should be implemented using socket/http].

Suggested Evaluation Methods:
- Demonstration and assessment of the implemented algorithms.
- Review web services implementation - Proper Connection should be established between the client and server to make use of the service offered by the Server.

UNIT II INTRODUCTION TO CLOUD COMPUTING 8
Suggested Activities:
- Explore public cloud services like Amazon, Google, Sales Force, Digital Ocean, Azure

Suggested Evaluation Methods:
- Quizzes on different service models and deployment models.
- Report submission - Comparison of various services provided by different Cloud public Service Providers (configuration of VM, cost, network bandwidth etc.).

UNIT III CLOUD ENABLING TECHNOLOGIES

Suggested Activities:
- Install Oracle Virtual Box/VMware Workstation and create a chat application [Note: Launch two virtual machines for chat application].
- Install Docker and create containers.

Suggested Evaluation Methods:
- Review the working of application in virtual environment.
- Evaluate the working of any application in the container created.

UNIT IV CLOUD MANAGEMENT, STORAGE AND SECURITY

Suggested Activities:
- Use security tools like OSSEC, ETTERCAP for finding security vulnerabilities.
- Deploy Open vSwitch or Mininet to explore cloud networking model.

Suggested Evaluation Methods:
- Report submission - Generate a detailed report describing vulnerabilities.

UNIT V CLOUD SOFTWARE AND COMPUTING PLATFORMS

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 through web browser. Verify the working of instance by logging into it/pinging the instance.

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:
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.

TOTAL: 60 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 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.

CA5304 ARTIFICIAL INTELLIGENCE & MACHINE LEARNING

OBJECTIVES:
- To understand the fundamentals of Search techniques
- To understand the reasoning methods in Intelligent systems
- 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.

UNIT I AGENTS AND SEARCH


Suggested Activities
- Flipped classroom on structure of agents.
- Flipped classroom on uninformed search - searching with costs.
- In-class activity on solving puzzles with uninformed and informed searches.

Suggested Evaluation Methods
- Assignments on puzzles with uninformed and informed searches.
- Quizzes on environments and search
- Evaluation of the programming exercises.

UNIT II LOGIC AND REASONING

Proposition Logic - Syntax - Semantics - First Order Logic - Syntax - Semantics - Conversion from English Statements to First order logic formula - Reasoning methods - Forward chaining - Backward chaining – Resolution – Introduction to Probabilistic Reasoning

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 III  BASICS OF MACHINE LEARNING
9

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 hmmptk and pgmpy.

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

UNIT IV  SUPERVISED LEARNING
9

Suggested Activities:
• Flipped classroom on basics about classification and regression.
• Practical - Collection of data from different resources 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 V  UNSUPERVISED LEARNING
9

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

Suggested Evaluation Methods:
• Assignments on mixture models.

TOTAL: 45 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
1. Apply the search techniques to real-time problems
2. Understand the underpinnings of Logic and Reasoning
3. Choose and implement classification or regression algorithms for an application using an open source tool.
4. Implement probabilistic discriminative and generative algorithms for an application and analyze the results.
5. Use a tool to implement typical clustering algorithms for different types of applications.
6. Implement appropriate learning algorithms for any real time application using an open source tool.

REFERENCES:

CA5305 INTERNET OF THINGS

OBJECTIVES:
- To understand the fundamentals of Internet of Things.
- To build a small low-cost embedded system using Arduino / Raspberry Pi or equivalent boards.
- To learn communication protocols that is frequently used in IoT ecosystems.
- To explore the ways of processing enormous amount of data generated in IoT based systems.
- To understand the role of cloud computing in IoT and to become familiar with various cloud offerings.

UNIT I ENABLING TECHNOLOGIES AND REFERENCE MODELS

Suggested Activities:
- Flipped classroom on enabling technologies.
- External learning – Exploring proprietary protocols used in IoT and M2M.
- Analyzing the required level of design for different IoT based ecosystems.

Suggested Evaluation Methods:
- Quiz and discussion on enabling technologies (WSN, Cloud and Big Data).
- Assignments on proprietary protocols used in IoT and M2M.
- Deciding the level and designing the IoT framework for case studies.

UNIT II DESIGN OF END DEVICES

Suggested Activities:
- Flipped classroom on open source movement in hardware and SDLC for embedded systems.
- Explore the variants of Arduino Boards, Atmel Microcontrollers, Cypress Pioneer and NXP Freedom.
- Learning to write Arduino Sketches and Python programs.
Suggested Evaluation Methods:
- Quiz and discussion on open source movement in hardware and SDLC for embedded systems.
- Assignments on Arduino boards, Atmel Microcontrollers, Cypress Pioneer and NXP Freedom.
- Practical – Developing Arduino Scripts and Python programs.

UNIT III  IoT PROTOCOLS
MAC Layer Protocols – IEEE 802.15.4 – G and E Variants of IEEE 802.15.4 – IEEE 802.11ah – IEEE 1901.2a – LoRaWAN – 6LoWPAN – From 6LoWPAN to 6Lo – NBIoT – REST Based Protocols – SCADA, CoAP and MQTT

Suggested Activities:
- External learning – Explore various software tools that support CoAP and MQTT.
- Flipped classroom on role of Ipv6 in designing IoT based systems.

Suggested Evaluation Methods:
- Assignments on software tools that support Coap and MQTT.
- Quiz and discussion on role of Ipv6 in IoT based systems.
- Assignments on the IoT policy of Meity (Government of India).

UNIT IV  INDUSTRIAL IoT
Industrial IoT adoption – IIoT Challenges, Drivers and Taxonomies – Industry 4.0- Areas of IIoT Adoption – Tools and Technologies assisting IIoT – Case studies, Retail Industry, Home automation, Manufacturing Automation, Energy management, Health care and Workflow Management.

Suggested Activities:
- External learning – Industry 5.0 (Exploring scalability, security, and customization).
- Flipped classroom on IIoT business Models.
- Learning and Imbibing Green IoT.

Suggested Evaluation Methods:
- Assignments on emerging areas of Industry 5.0.
- Quiz and group discussion on technologies for IIoT business Models.
- Practicing green framework for IoT application (expeditionary learning-project based)

UNIT V  IoT ANALYTICS

Suggested Activities:
- Flipped classroom on cloud models and type of clouds.
- External learning – Django framework.

Suggested Evaluation Methods:
- Quiz and discussion on cloud models and types of clouds.
- Developing web apps for IoT ecosystems using Django framework.

PRACTICAL EXERCISE:
1. Develop a BLINK sketch in Arduino.
2. Develop an Arduino sketch that repeats an LED to glow brightly, decrease the brightness, switches off the LED, increases the brightness and LED glows with maximum intensity (a sketch for fading).
3. Develop an Arduino sketch that takes sensor readings for five seconds during the startup and tracks the highest and lowest values it gets. These sensor readings during the first five
seconds of the sketch execution define the minimum and maximum of expected values for the readings taken during the loop (a sketch for calibrating a sensor).
4. Develop an Arduino sketch that reads the value of a variable resistor as an analog input and changes blink rate of the LED.
5. Develop an Arduino sketch to use a piezo element to detect the vibration.
6. Develop a Python program to control an LED using Raspberry Pi.
7. Develop a Python program to interface an LED with a switch using Raspberry Pi.
8. Implement a map reduce program that produces a weather data set.
9. Implement an application that stores big data in Hbase/Mongo DB using Hadoop/R.
10. Use Google Collaboration Tools: Create Google Docs, Sheets, and Slides and share it with others.
11. Mini project.

TOTAL: 75 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
1. Understand the enabling technologies and reference models of IoT.
3. Apply appropriate protocols in various parts of IoT based systems.
4. Understand Big Data tools and technologies and apply them in IoT based systems.
5. Design and deploy IoT based systems and connect them to cloud offerings.
6. Design IoT systems for various real time applications.

REFERENCES:

CA5306 CYBER 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

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 standards.

UNIT III PLANNING FOR CONTINUITY

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

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, analysisflaw 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 the information about the 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 analysis various attacks like buffer overflow, XSS etc.

EXPERIMENTS

1. Study of System threat attacks - Denial of Services.
2. Study of Sniffing and Spoofing attacks.
3. Study of Techniques uses for Web Based Password Capturing.
4. Study of Different attacks causes by Virus and Trojans.
5. Study of Anti-Intrusion Technique – Honey pot.
6. Implement the following SUBSTITUTION & TRANSPOSITION TECHNIQUES concepts:
   a. Caesar Cipher
   b. Playfair Cipher
   c. Hill Cipher
   d. Vigenere Cipher
   e. Rail fence – row & Column Transformation
7. Implement the following algorithms
   a. DES
   b. RSA Algorithm
   c. Diffie-Hellman
   d. MD5
   e. SHA-1
8. Implement the Signature Scheme - Digital Signature Standard
9. Demonstrate how to provide secure data storage, secure data transmission and for creating digital signatures (GnuPG)
10. Setup a honey pot and monitor the honeypot on network (KF Sensor)
11. Installation of rootkits and study about the variety of options
12. Perform wireless audit on an access point or a router and decrypt WEP and WPA.( Net Stumbler)
13. Demonstrate intrusion detection system (ids) using any tool (snort or any other s/w)

TOTAL: 75 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
1. Apply the basic security models and policies required by computing system.
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:
CA5314  MACHINE LEARNING TECHNIQUES LABORATORY

OBJECTIVES:
- To apply the concepts of Machine Learning to solve real-world problems
- To implement basic algorithms in clustering & classification applied to text & numeric data
- To implement algorithms related to dimensionality reduction

EXERCISES
1. Linear Regression and Multiple Regression
2. K-Nearest Neighbor Classifier
3. Root Node Attribute Selection for Decision Trees Using ID3
4. Solving Regression & Classification using Decision Trees - Cart
5. Using Weka Tool for SVM Classification for Chosen Domain Application
6. Data & Text Clustering Using K-Means Algorithm
7. Dimensionality Reduction Algorithms in Image Processing Applications
8. Naive Bayes Classifier
9. Random Forest

TOTAL: 60 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
1. Understand and apply unsupervised learning algorithms for clustering.
2. Implement machine learning algorithms related to numeric data
3. Learn the application of supervised machine learning algorithms
4. Use dimensionality reduction algorithms for image processing applications
5. Use fundamental regression algorithms for solving real-world data
6. Decide to apply a specific type of ML algorithm for real-world problems

CA5001  BLOCKCHAIN TECHNOLOGIES

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  9

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  9

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  9

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  9

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.

TOTAL: 45 PERIODS

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:

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 problems on information gathering and traceroute of ICMP, DNS and SNMP.
- 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:
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.

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.

TOTAL: 45 PERIODS

REFERENCES:
CA5007 E-LEARNING TECHNIQUES

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  9

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  9

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  9

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  9
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  9

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.

TOTAL: 45 PERIODS

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:

CA5008 SOFTWARE TESTING  L  T  P  C
3  0  0  3

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
Defect Prevention Strategies.

**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**


**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.

**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.

**TOTAL: 45 PERIODS**

**REFERENCES:**

**CA5009 DEEP LEARNING TECHNIQUES AND APPLICATIONS**

**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 \hspace{1cm} BASICS OF NEURAL NETWORKS \hspace{1cm} 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 \hspace{1cm} INTRODUCTION TO DEEP LEARNING \hspace{1cm} 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 \hspace{1cm} CONVOLUTIONAL NEURAL NETWORKS \hspace{1cm} 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 \hspace{1cm} ADDITIONAL DEEP LEARNING ARCHITECTURES \hspace{1cm} 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


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

TOTAL: 45 PERIODS

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:

CA5010 GAME PROGRAMMING TECHNIQUES

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.
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.

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:

CA5011 MULTIMEDIA TECHNOLOGIES

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.
• Practical - Adobe Premier Pro for Digital Video Concepts.
• External learning - Adobe aftereffects, Adobe Media Encoder and Adobe Audition.

Suggested Evaluation Methods:
• Demonstration, finalization and output of the practical learning.
• Quizzes on MPEG and audio encoding.

UNIT III MULTIMEDIA ARCHITECTURES

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

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

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:

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 9

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.
UNIT III DATA PRESENTATION 9

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 9

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 9

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:

CA5014 C# AND .NET PROGRAMMING L T P C
3 0 0 3

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
Diagnostics Tasks – Threads and Synchronization – Manipulating XML – SAX and DOM –
Manipulating files and the Registry – Transactions – Data access with ADO.NET: Introduction, LINQ to Entities and the ADO.NET Entity Framework, Querying a Database with LINQ – Creating the ADO.NET Entity Data Model Class Library, Creating a Windows Forms Project – Data Bindings Between Controls and the Entity Data Model – Dynamically Binding Query Results.

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


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


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.

REFERENCES:
3. Ian Gariffiths, Mathew Adams, Jesse Liberty, “Programming C# 4.0”, O’Reilly, Sixth Edition,
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

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

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

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

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

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:
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  9

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  9

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.

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:

CA5017  MIXED REALITY  L T P C  3 0 0 3

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:
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
9

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 9

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 9

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:
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

UNIT II  TEXT CATEGORIZATION AND CLUSTERING

UNIT III  TEXT MINING FOR INFORMATION RETRIEVAL AND EXTRACTION

UNIT IV  PROBABILISTIC MODELS
Probabilistic Models for Text Mining – Mixture Models – Stochastic Processes in Bayesian

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:

CA5020 DATA WAREHOUSING AND DATA MINING TECHNIQUES

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:

CA5021 SOFTWARE QUALITY ASSURANCE

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


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


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


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.

TOTAL: 45 PERIODS
6. Document the testing procedures.

REFERENCES:

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
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
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

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

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:

CA5028 WIRELESS SENSOR NETWORKS & PROTOCOLS

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

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

**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:

CA5032 SEMANTIC WEB AND APPLICATIONS

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:

CA5033                      SOFT COMPUTING                  L  T  P  C
                                      3 0 0 3

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

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  

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  

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  COMPETITIVE NEURAL NETWORKS  
Kohenen’s Self Organizing Map – SOM Architecture, learning procedure – Application; Learning Vector Quantization – learning by LVQ; Adaptive Resonance Theory – Learning procedure – Applications.

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:

OBJECTIVES:
- To know the theoretical background of cognition.
- To understand the link between cognition and computational intelligence.
- To explore probabilistic programming language.
- To study the computational inference models of cognition.
- To study the computational learning models of cognition.

UNIT I PHILosophy, PSYchology AND NEUROscience 9

Suggested Activities:
- Flipped classroom on logic and sciences in the mind.
- Case study on how philosophy (western and eastern), psychology and neuroscience (thought process in normal persons, children and differently abled) helps in cognition.
- Mind map of cognition with various attributes such as mind, logic, information processing etc.
- Discussion and debate on cognition.

Suggested Evaluation Methods:
- Quiz on the topic logic and sciences in the mind.
- Active discussion on the case study of how the factors such as learning and memory affect cognition.
- Essay writing on how various factors influence cognition.
UNIT II  COMPUTATIONAL INTELLIGENCE


Suggested Activities:
- Flipped classroom on knowledge-based systems.
- Mind map on different methods of cognition in computational domain.
- Discussion on the influence of human cognition systems with a link to computational domain.

Suggested Evaluation Methods:
- Quiz on knowledge-based systems.
- Collaborative wiki editing of computational tools linking with cognition.
- Essay writing on the computational cognitive systems with the background of human cognitive systems.

UNIT III  PROBABILISTIC PROGRAMMING LANGUAGE


Suggested Activities:
- Flipped classroom on JavaScript libraries.
- Exploration of the existing mathematical models.
- Practical Programming the common mathematical functions using PPL.

Suggested Evaluation Methods:
- Quiz on the basics of JavaScript and WebPPL.
- Practical Programming assignment on developing miniature programs using WebPPL for inference mechanisms.
- Evaluation of the programming assignments.

UNIT IV  IMPLEMENTING THE INFERENCE MODELS OF COGNITION


Suggested Activities:
- Flipped classroom on casual and statistical dependence.
- Perform sample calculation of different inference models manually.

Suggested Evaluation Methods:
- Quiz on statistical dependence
- Practical Automation of the mathematical functions through WebPPL.
- Practical Programming assignments on analyzing data through cognitive models with webPPL.
- Evaluation of the programming assignments.

UNIT V  IMPLEMENTING THE LEARNING MODELS OF COGNITION


Suggested Activities:
- Flipped classroom on Mixture models.
- Perform sample calculation of models manually.

Suggested Evaluation Methods:
- Quiz on Mixture models.
- Practical – Automation of the mathematical functions through WebPPL.
- Practical - Programming assignment on learning models for continuous functions.
- Evaluation of the programming assignments.

TOTAL: 45 PERIODS

OUTCOMES:
On completion of the course, the student will be able to:
1. Understand the underlying theory behind cognition.
2. Connect to the cognition elements computationally.
3. Implement mathematical functions through WebPPL.
4. Develop a cognitive inference model.
5. Develop a cognitive learning model.
6. Explore the recent trends in cognitive computing.

REFERENCES:

OBJECTIVES:
- To provide a sound knowledge in various middleware technologies
- To understand the middleware usage in distributed environment
- To gain the knowledge about RMI, CORBA and EJB
- To understand the development of web applications using SOAP and other web services
- To familiarize between various web service architectures and their standards
- To provide the knowledge to develop middleware application for real time usage

UNIT I INTRODUCTION

Suggested Activities:
- Development of distributed applications using RMI.
- Development of synchronous and asynchronous Java based Messaging Services.

Suggested Evaluation Methods:
- Quiz on RMI and distributed services.
- Demonstration of RMI and web services implementation.

UNIT II EJB and CORBA

Suggested Activities:
- Development of distributed applications using CORBA and Java Bean.
• Development of services using EJB.

Suggested Evaluation Methods:
• Quiz on CORBA and EJB based web services.
• Demonstration of EJB and CORBA web services implementation.

UNIT III  COM and .NET  9
Evolution of DCOM - Introduction to COM - COM clients and servers - COM IDL - COM Interfaces
COM Threading Models – Marshalling - Custom and standard marshalling - Comparison COM and
CORBA - Introduction to .NET - Overview of .NET architecture - Remoting.

Suggested Activities:
• Developing applications using COM and .NET.
• Flipped classroom on web services with .NET.

Suggested Evaluation Methods:
• Quiz on COM and .NET services.
• Demonstration of COM based web application using .NET.

UNIT IV  SOA and WEB SERVICES  9
Defining SOA - Business value of SOA - SOA characteristics - Concept of a service, Basic SOA -
Enterprise Service Bus (ESB) - SOA enterprise Software Models -Services and SOA – WSDL -
SOAP, UDDI, WS Standards -Web Services and Service Oriented Enterprise (SOE) - Coordination

Suggested Activities:
• Creation of a SOAP and RESTful based web services.
• Creation of services using UDDI and WSDL

Suggested Evaluation Methods:
• Assignment on various Case studies to SOA applications.
  Demonstrations on the applications developed and its comparison.

UNIT V  OTHER TYPES OF MIDDLEWARE
Other types of Middleware, Real-Time Middleware, Embedded Systems Middleware, Mobile
Middleware, Oracle Fusion Middleware.

Suggested Activities:
• Analysis of the applications of various Middleware.
• Creation of any web based middleware application.

Suggested Evaluation Methods:
• Assignment on case studies of various Middleware applications.
• Demonstration of the web based middleware applications and its comparison

OUTCOMES:
On completion of the course, the student will be able:
1. To implement the distributed services using RMI and CORBA
2. To implement programs in EJB
3. To map and differentiate the functions between COM and .NET
4. To outline the functionalities of various types of middleware technologies
5. To design web services using SOAP, UDDI, WSDL
6. To design the middleware applications for real time usage.

REFERENCES:
1. G. Sudha Sadasivam, Radha Shankarmani, “Middleware and Enterprise Integration

CA5036 UI & UX

OBJECTIVES:
- To provide a sound knowledge in UI & UX
- To understand the need for UI and UX
- Research Methods used in Design
- Tools used in UI & UX
- Creating a wireframe and prototype

UNIT I FOUNDATIONS OF DESIGN
UI vs. UX Design - Core Stages of Design Thinking - Divergent and Convergent Thinking - Brainstorming and Game storming - Observational Empathy

Suggested Activities:
- Hands on Design Thinking process for a product
- Brainstorming features for a product

Suggested Evaluation Methods:
- Evaluate final product of design thinking
- Evaluate result for a product

UNIT II FOUNDATIONS OF UI DESIGN
Visual and UI Principles - UI Elements and Patterns - Interaction Behaviors and Principles – Branding - Style Guides

Suggested Activities:
- Defining the Look and Feel of any new Project
- Create a Sample Pattern Library for that product (Mood board, Fonts, Colors based on UI principles)

Suggested Evaluation Methods:
- Evaluate the designs based on UI principles

UNIT III FOUNDATIONS OF UX DESIGN

Suggested Activities
- Identify a customer problem to solve - Main Project start 1

Suggested Evaluation Methods
- Customer problem statement should match any of the main problem faced in real time
UNIT IV RESEARCH, DESIGNING, IDEATING, & INFORMATION ARCHITECTURE


Suggested Activities
- Conduct end-to-end user research - User research, creating personas, Ideation process (User stories, Scenarios), Flow diagrams, Flow Mapping - Main Project continues 2

Suggested Evaluation Methods
- Evaluate the user research by user story and scenarios

UNIT V WIREFRAMING, PROTOTYPING AND TESTING


Suggested Activities
- Sketch, design and build a prototype and perform usability testing and identify improvements - Main project ends 3

Suggested Evaluation Methods
- Evaluate wireframe by usability
- Evaluate prototype by UI principles and usability

TOTAL: 45 PERIODS

OUTCOMES:
On Completion of the course, the students should be able to:
1. Build UI for user Applications
2. Know the UI Interaction behaviors and principles
3. Evaluate UX design of any product or application
4. Demonstrate UX Skills in product development
5. Implement Sketching principles
6. Create Wireframe and Prototype

REFERENCES:
2. Steve Schoger, Adam Wathan “Refactoring UI”, 2018
4. https://www.interaction-design.org/literature

CA5037 ROBOTIC PROCESS AUTOMATION

OBJECTIVES:
- To understand Basic Programming concepts and the underlying logic/structure
- To Describe RPA, where it can be applied and how its implemented
- To Describe the different types of variables, Control Flow, and data manipulation techniques
- To Understand Image, Text, and Data Tables Automation
- To Describe automation to Email and various types of Exceptions and strategies to handle

UNIT I RPA CONCEPTS

RPA Basics - History of Automation - What is RPA - RPA vs Automation - Processes & Flowcharts - Programming Constructs in RPA - What Processes can be Automated - Types of Bots - Workloads which can be automated - RPA Advanced Concepts - Standardization of processes - RPA

Suggested Activities
- See an RPA solution in production
- Analyse a case study for RPA and its lifecycle
- Compare tools and know the differences
- Identify companies using RPA solution and discuss the use cases

Suggested Evaluation Methods
- Build a use case for RPA and its lifecycle
- Difference between RPA and Test Automation
- Components Understanding
- Architecture Understanding
- Tools Differentiation
- Difference between RPA and AI

UNIT II RPA TOOL INTRODUCTION & BASICS


Suggested Activities
- Implement Variables, Data Types and Control Flow
- Apply Data Manipulation techniques

Suggested Evaluation Methods
- Building scenarios to apply learning from this unit

UNIT III AUTOMATION CONCEPTS INTRODUCTION

Recording and Advanced UI Interaction - Recording Introduction - Basic and Desktop Recording - Web Recording - Input/Output Methods - Screen Scraping - Data Scraping - Scraping advanced techniques - Selectors - Selectors - Defining and Assessing Selectors - Customization - Debugging - Dynamic Selectors - Partial Selectors

Suggested Activities
- Implement Recording and UI Interactions
- Implement Selectors and Keyboard based automation

Suggested Evaluation Methods
- Building scenarios to apply learning from this unit

UNIT IV ADVANCED AUTOMATION CONCEPTS AND TECHNIQUES

**Suggested Activities**
- Implement Recording and UI Interactions
- Implement Selectors and Keyboard based automation
- Implement Scraping techniques

**Suggested Evaluation Methods**
- Building scenarios to apply learning from this unit

**UNIT V  EMAIL AUTOMATION & EXCEPTIONAL HANDLING  6**
Email Automation - Email Automation - Incoming Email automation - Sending Email automation - Debugging and Exception Handling - Debugging Tools - Strategies for solving issues - Catching errors.

**Suggested Activities**
- Create and Debug errors in the workflow
- Read and extract content from PDF, Email
- Develop a fully functional Bot and share it with others

**Suggested Evaluation Methods**
- Building a fully functional Bot for a UseCase

**OUTCOMES**
On completion of the course, the students should be able to:
1. Understand the need and use of Automation
2. Describe RPA, where it can be applied and how it is implemented
3. Describe the different types of variables, Control Flow, and data manipulation techniques
4. Identify and understand Image, Text, and Data Tables Automation
5. Describe automation to Email and various types of Exceptions and strategies to handle
6. Build Bots which can do automation

**REFERENCES**
1. Alok Mani Tripathi, Learning Robotic Process Automation, Publisher: Packt Publishing
5. https://www.guvi.io/rpa

**OBJECTIVES:**
- To gain knowledge about various software development lifecycle (SDLC) models, software development techniques and its application in real world context.
- To understand process, process improvement, requirements engineering and requirements management.
- To be aware of designing a software considering the various perspectives of end user. To learn to develop a software component using coding standards and facilitate code reuse.
- To use advanced software testing techniques.
- To analyze the software using metrics and to predict the complexity and the risk associated with projects.
UNIT I  FORMAL METHODS AND AGILE METHODOLOGIES

Suggested Activities:
- Identify and propose ways to build quality software to stop deterioration due to change.
- Assignments like the following: Giving reasons for your answer based on the type of system being developed, suggest the most appropriate generic software process model that might be used as a basis for managing the development of the following systems (not limited to):
  - A system to control anti-lock braking in a car.
  - A virtual reality system to support software maintenance.
  - A university accounting system that replaces an existing system.
  - An interactive travel planning system that helps users plan journeys with the lowest environment impact.
- Using the FDD feature template, define a feature set for a web browser.

Suggested Evaluation Methods:
- Assignments on the selection of suitable software process models for a given software specification.
- Tutorial – Identification of sample application for each process model and justification of the same stating reasons.

UNIT II  REQUIREMENTS ENGINEERING AND MANAGEMENT

Suggested Activities:
- External Learning – Using open-source tools for RE to understand the requirements traceability and interdependency among the functionalities provided by the software project.
- External Learning – Requirements elicitation mechanisms and selection of an appropriate strategy.

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.
- Assignment on Selection of suitable software process models for a given software specification.

UNIT III  SOFTWARE DESIGN, OOAD AND SOFTWARE IMPLEMENTATION

Suggested Activities:
- Draw the activity network representation of the tasks.
- Determine ES, EF, and LS, LF for every task.
- Develop the Gantt chart representation for the project.
Suggested Evaluation Methods:
- Assignments on Software Design and modelling.
- Tutorial problems on UML Modelling.
- Quiz on Software Design methods and its implementation.

UNIT IV SOFTWARE PRACTICES, PROCESSES AND ARCHITECTURE

Suggested Activities:
- In-class activity on application specific product and process view.
- Develop a complete process framework for any project like healthcare system.
- A class project may be given as follows: Develop a debugging plan that will provide language and system-oriented hints that have been learnt. Begin with the outline of topics that will be revived by the class and your instructor. Publish the debugging plan for others in your local environment.

Suggested Evaluation Methods:
- Assignment on testing sample application and understand the differences in selecting of test cases from the test suite.
- Tutorial problems on software development methodologies.
- Quiz on software development methodologies and software engineering practices.

UNIT V SOFTWARE PROJECT MANAGEMENT

Suggested Activities:
- Perform software project management for any project like automobile in which the driver commands the steering wheel and provides vision.
- Develop a schedule timeline.
- Develop Mini projects (Software Project Categories – PHP Projects, Data Mining, Android Projects, Smart Card/ Biometrics, Dotnet Projects, Web Based Projects, Information Security, IOS Projects, Artificial Intelligence, Embedded Projects)

Suggested Evaluation Methods:
- Assignment on testing sample application and understanding the differences in the selection of test cases from the test suite.
- Tutorial problems on risk management, configuration management, quality management, planning and scheduling.
- Quiz on risk management, configuration management, quality management, planning and scheduling.

TOTAL: 45 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
1. Analytically apply general principles of software development in the development of complex software and software-intensive systems.
2. Understand methods and techniques for advanced software development and also be able to use these in various development situations.
3. Apply testing techniques for object-oriented software and web-based systems.
4. Familiarize with the basic concepts of Software design and implementation.
5. Apply various software metrics on software quality products.
6. Apply various skills on real-time projects.
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 describe the role of Unmanned Aerial Vehicles (UAVs) Drones system.
- To understand and describe basic regulations applicable to UAV flight.

UNIT I
INTRODUCTION TO AUTONOMOUS DRIVING
9

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
9

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 OVERVIEW OF UNMANNED AERIAL VEHICLES

Suggested Activities:
- Flipped classroom on categories of Drones.
- External learning – Team Daksha from MIT.
- Assignment on the working principles of various Drones.

Suggested Evaluation Methods:
- Quizzes on policies for using UAVs in Home and Foreign Regularities.
- Viva Voce on assignment topics.
- Practical – Safety consideration of flying Drones.

UNIT V METHODOLOGIES IN FLYING UAVs

Suggested Activities:
- Flipped classroom on UAVs flight control.
- External learning - Study on ground station control in airports.
- Assignment - Communication protocols for UAVs.

Suggested Evaluation Methods:
- Viva Voce on assignment topic.
- Practical - Experiment on simple axes controls.
- Practical - Experiment on sensing and surveillance.

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. Know about the policies in flying UAVs.
5. Ethics in using UAVs according to regulations.
6. Design communication protocols for UAVs from ground stations controls.

REFERENCES:

CA5040 FINANCIAL TECHNOLOGIES
3 0 0 3

OBJECTIVES:
- To learn the fundamentals of the financial technologies, and Investment analysis
- To study the different ways of raising money with Fintech
- To learn how to harness data with artificial intelligence and machine learning
- To understand and describe the digital finance and alternative finance and the future of Fintech.
- Developing the capability in terms of the applications of tools and techniques in analyzing and solving problems related to investment

UNIT I FINTECH 9

Suggested Activities:
- Flipped classroom on basics of FinTech Typology.
- External learning - The future of RegTech and the Technologies impacting it.

Suggested Evaluation Methods:
- Quizzes on FinTech.
- Assignments on Emerging Economics.

UNIT II INNOVATIONS OF FINTECH 9
Conclude whether the disintermediation of banks in the provision of credit is a transitory or permanent phenomenon. Individual Payments, Mobile Money, Regulation of Mobile Money, RTGS Systems, ABCDs of Alternative Finance, Building a New stack Cryptocurrencies, Legal and Regulatory Implications of Cryptocurrencies, Blockchain, The Benefits from New Payment Stacks (Applications of Ripple)

Suggested Activities:
- Flipped classroom on basics of Mobile Money.
- External learning - Legal and Regulatory Implications of Cryptocurrencies

Suggested Evaluation Methods:
- Quizzes on ABCDs of Alternative Finance.
- Demonstration of the practical implementations of Blockchains.
UNIT III FINTECH REGULATIONS AND DATA REGULATION

FinTech Regulations - Evolution of RegTech - RegTech Ecosystem: Financial Institutions, Startups, Challenges, Regulators, Regulatory Sandboxes, Smart Regulation. Data Regulation Data in Financial Services-European Big-Bang: PSD2 / GDPR / MiFID2 – Digital Identity- Regulation 1.0 to 2.0 (KYC to KYD)

Suggested Activities:
- Flipped classroom on basics of data regulations of various countries.
- External learning - Transforming Personal Finance with FinTech
- External learning - Application of AI in Smart Regulation (Mindbridge)
- Read: https://www.predictiveanalyticstoday.com/artificial-intelligence-platforms/

Suggested Evaluation Methods:
- Quizzes on FinTech Regulations.
- Assignments on automation in the investment management industry, Balancing Innovation and Regulation Challenges.
- Industry Showcase: Cybersecurity Industry, PSD2: Open Banking API for Startups (Gini), Application of Data Analytics in Finance.

UNIT IV DIGITAL FINANCE AND ALTERNATIVE FINANCE


Suggested Activities:
- Flipped classroom on basics of Digitization of Financial Services.
- External learning - Ensuring Compliance from the Start

Suggested Evaluation Methods:
- Quizzes on Financial Innovation.
- Assignments on Method of AI used to Transform the Future of FinTech.

UNIT V BUILDING & MANAGING A SUCCESSFUL FINTECH STARTUP

Understanding the impact of Macro & Micro factors on the Business Dynamics- Art & Science of Design Thinking Managing Growth, Fund Raising and Exits. Disruptive Technology Cases in FinTech

Suggested Activities:
- Flipped classroom on basics of TechFin.
- External learning – How is FinTech reconfiguring financial services business models? What are the key disruption points? What determines success in FinTech?
- External learning – Where are the limits, risks, and broader policy and social implications of FinTech?

Suggested Evaluation Methods:
- Assignments on automation in the investment management industry.
- Case Studies- Revolut, Alibaba, Aadhaar, Credit Karma, Digibank.

OUTCOMES:
On successful completion of the course, the students will be able to:
1. Apply the concepts and computational basics in the real-world financial market scenario
2. Formulate trading strategies by identifying the patterns in trading and market price movements
3. Evaluate portfolios through systematic technical and fundamental analysis

TOTAL:45 PERIODS
4. Collaborate and compete with trading groups in a simulated environment and extend to the real investment scenarios
5. Demonstrate decision dynamics to attain the investment objectives in a stock market environment
6. Learn to assess the future of fintech and think strategically about challenges faced by financial companies

REFERENCES

CA5041 LINUX ADMINISTRATION

OBJECTIVES
● To explore the commands for accessing hardware resources
● To learn the concepts of Linux file systems
● To understand the working of boot process, kernel, and user spaces
● To explore different process and memory management tasks
● To familiarize the network configuration files

UNIT I INTRODUCTION

Suggested Activities
● Flipped classroom on internals of any Operating System
● In class activity - overall view of Linux installation

Suggested Evaluation Methods
● Quizzes
● Programming assignments on shell commands

UNIT II DEVICES, DISKS and FILE SYSTEMS

Suggested Activities
● Flipped classroom on various devices
● In class activity - Exploring the devices through commands in Linux environment

Suggested Evaluation Methods
● Quizzes
● Analysis of the device usage
● Programming with shell commands to understand the usage of various devices

UNIT III KERNEL SPACE AND USER SPACE
How the Linux Kernel Boots: Startup messages – Kernel initialization and Boot options – Kernel

Suggested Activities
- Flipped classroom on the booting methods and file systems
- In class activity - Exploring the location and function of bootloaders in Linux

Suggested Evaluation Methods
- Quizzes
- Dual boot using GRUB
- Exploring bootloader commands

UNIT IV SYSTEM CONFIGURATION, PROCESS AND RESOURCE UTILIZATION

Suggested Activities
- Flipped classroom on the types of logs and CPU/ memory management
- In class activity - Exploring the usage of CPU and memory through commands

Suggested Evaluation Methods
- Quizzes
- Time analysis to process different types of programs
- Memory usage analysis for different process using shell commands

UNIT V NETWORK CONFIGURATION AND SERVICES

Suggested Activities
- Flipped classroom on network concepts
- In class activity - Exploring the network details through commands

Suggested Evaluation Methods
- Quizzes
- Editing configuration files for different network setup
- Usage of different ssh commands to access a remote machine

OUTCOMES:
On completion of the course, the students will be able to:
1. Understand an overall view of the structure of Linux
2. Access the different devices through commands
3. Work with kernel and user spaces in Linux environment
4. Automate tasks using scheduling tools
5. Configure network files based on the specific need
6. Acquire Linux Administration skills to manage a server

TOTAL: 45 PERIODS
REFERENCES:

CA5042  VOICE TECHNOLOGY  L T P C
3 0 0 3

OBJECTIVES:
- To understand the fundamentals of the speech processing
- Explore the various speech models
- Gather knowledge about the phonetics and pronunciation processing
- Perform wavelet analysis of speech
- To understand the concepts of speech recognition

UNIT I  VOICE COMPUTING  9

Suggested Activities:
- Flipped classroom on basics of voice computing.
- External learning - mixers
- Practical - Implementation of any one application of Text to Speech systems.

Suggested Evaluation Methods:
- Quizzes on voice computing.
- Assignments on Reading and writing of audio files.
- Demonstration of the practical implementations of Text to Speech systems.

UNIT II  FEATURES  9

Suggested Activities:
- Flipped classroom on basics of features.
- External learning - Automatic speech recognition

Suggested Evaluation Methods:
- Quizzes on features.
- Assignments on Dimensionality Reduction.
- Demonstration of the practical implementations of Automatic speech recognition.

UNIT III  MODELING  9

Suggested Activities:
- Flipped classroom on basics of Classification and Regression models.
- External learning - Speech Modelling
- Practical - Implementation of any one application of Phonetics.

Suggested Evaluation Methods:
- Quizzes on Classification and Regression models.
• Assignments on Classification and Regression models.

UNIT IV MACHINE VOICE

Suggested Activities:
• Flipped classroom on basics of Speech to text and Text to Speech systems.
• External learning - speech recognition by humans

Suggested Evaluation Methods:
• Quizzes on acoustic models.
• Assignments on Speech to text and Text to Speech systems.
• Demonstration of the practical implementations.

UNIT V VOICE USER INTERFACE

Suggested Activities:
• Flipped classroom.
• Practical - Building Google Assistant

Suggested Evaluation Methods:
• Quizzes on Voice UI.
• Assignments on Speech synthesis Mark-up Language.
• Demonstration of the practical implementations of Voice user Interface.

TOTAL:45 PERIODS

OUTCOMES:
On Successful completion of the course, Students will be able to
1. Understand the basics of Voice computing
2. Derive Features form audio, speech, and voice.
3. Understand the need for machine learning and basic of HMM models
4. Understand the need for speech identification system
5. Generate a new speech recognition system
6. Gain a knowledge of Voice applications for Alexa and Google Assistant

REFERENCES:
1. Daniel Jurafsky and James H. Martin, Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition, Person education, 2013.
OBJECTIVES:
- To get exposed to the concepts of Business Domains and verticals.
- To understand the concepts in the manufacturing & service sector.
- To portray the modern retailing.
- To develop a conceptual and analytical understanding of the information systems.
- To understand the concepts of Portfolio Theory and Practice.

UNIT I  BUSINESS DOMAINS & VERTICALS

Suggested Activities:
- Flipped classroom on basics of Business Domains.
- External learning - Business domain with respect to mission, incorporation, organization.
- Practical - Implementation of any one prototype of one verticals.

Suggested Evaluation Methods:
- Quizzes on various domains.
- Case study on business domains and verticals.

UNIT II  MANUFACTURING & SERVICE SECTOR

Suggested Activities:
- Flipped classroom on basics of service sector.
- External learning - HRM and marketing Function.
- Practical - Implementation of any one application of Service Sector.

Suggested Evaluation Methods:
- Explore the analytical applications to Manufacturing and service sector.
- Case study on ICT for the service sector.

UNIT III  MODERN RETAILING
Role of Analytics in Retail sector – Retail Analytics Framework - Retailing Marketplace, Market space and Understanding Technological Aspects - Overview on R/R-Studio - Data access and Basic Analysis using R

Suggested Activities:
- Flipped classroom on basics of Retail Analytics.
- External learning – Role of Analytics in Retail sector.
- Practical - Implementation of any one application of - Data access and basic analysis using R.

Suggested Evaluation Methods:
- Quizzes on Retail Analytics.
- Assignments on Data access and basic analysis using R.

UNIT IV  INFORMATION SYSTEMS FOR BUSINESS
Suggested Activities:
- Flipped classroom on basics of MIS.
- External learning - Analytics and Business Intelligence
- Practical - Implementation of any one application of Enterprise Business Applications.

Suggested Evaluation Methods:
- Quizzes on ERPs.
- Assignments on MIS.
- Demonstration of the practical implementations Analytics for various business domains.

UNIT V PORTFOLIO THEORY AND PRACTICE
Portfolio Management; Markowitz portfolio Theory; portfolio mathematics; portfolio return; portfolio risk; capital allocation; optimal risky portfolios; index models; Equity portfolio management strategies; Overview of style analysis; asset allocation strategies; Evaluation of Portfolio performance – Composite Portfolio Performance measures; Application of Portfolio performance measures; Evaluation of bond portfolio performance.

Suggested Activities:
- Flipped classroom on basics of Portfolio.
- External learning - portfolio return; portfolio risk; capital allocation; optimal risky portfolios
- Practical - Implementation of any one application of Portfolio performance measures.

Suggested Evaluation Methods:
- Quizzes on Portfolio Management.
- Demonstration of the practical implementations.

TOTAL:45 PERIODS

OUTCOMES:
On Successful completion of the course, Students will be able to
1. Fundamentals of Business domains & verticals
2. Overview of manufacturing & service sector
3. Role of analytics in retail sector
4. Portfolio theory and practice
5. Importance of Information Systems as business support tool

REFERENCES:
2. Ramesh Sharda, Dursun Delen and Efraim Turban, “Business Intelligence and Analytics: Systems for Decision Support”, Tenth edition, Pearson, 2018

CA5044 DIGITAL FORENSICS

OBJECTIVES:
- To provide an understanding Computer Forensics fundamental.
- 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 video.

UNIT I INCIDENT AND INCIDENT RESPONSE


Suggested Activities:
- Survey of forensics tools such as WinHex, EnCase, FTK, or ProDiscover.
- External learning - 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:
- Demonstration on forensic tools
- Assignment on solving with sample cybercrime reports.

UNIT II FILE STORAGE AND DATA RECOVERY


Suggested Activities:
- Flipped classroom and activity.
- External learning - Tools for data storage and access, bypassing filtered [blocked] ports, reviewing Internet activity, open source forensic tools for file storage and data recovery will be introduced.

Suggested Evaluation Methods:
- Total quantity of files recovered from the disk for reconstruction.
- Quiz on forensic analysis of file system.

UNIT III NETWORK AND EMAIL FORENSICS


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

Suggested Evaluation Methods:
- Demonstration of Port Redirection tools.
- Practical - Assessment of real-time problems like email analysis for tracing.

UNIT IV SYSTEM FORENSICS


Suggested Activities:
- Demonstration of MD5Hash tool.
- Practical - IE Activity analysis.
Suggested Evaluation Methods:
- Assignment on live windows and Linux investigation
- Quiz on ethical hacking.

UNIT V  IMAGE AND VIDEO FORENSICS  9

Suggested Activities:
- External learning - Steganography.
- Practical – Install and use Steganalysis tool.

Suggested Evaluation Methods:
- Assignment on forgery detection in images.
- Quiz on locating and recovering Graphics files.

TOTAL:45 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
1. Recognize attacks on systems.
2. Design a counterattack 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 evidence of a computer crime.
6. Analyze various image encryption/decryption, steganography, and fraud in image.

REFERENCES:

OPEN ELECTIVE COURSES (OEC)

OE5091  BUSINESS DATA ANALYTICS  L  T  P  C
3  0  0  3

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  9
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 INFERENC

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:
1. Identify the real-world business problems and model with analytical solutions.
2. Solve analytical problem with relevant mathematics background knowledge.
3. Convert any real-world decision-making problem to hypothesis and apply suitable statistical testing.
4. Write and Demonstrate simple applications involving analytics using Hadoop and MapReduce
5. Use open source frameworks for modeling and storing data.
6. 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
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

UNIT III WEAR AND CORROSION AND THEIR PREVENTION

UNIT IV FAULT TRACING

UNIT V PERIODIC AND PREVENTIVE MAINTENANCE

OUTCOMES:
Students will be able to:
1. Ability to summarize basics of industrial safety
2. Ability to describe fundamentals of maintenance engineering
3. Ability to explain wear and corrosion

TOTAL: 45 PERIODS
4. Ability to illustrate fault tracing
5. Ability to identify preventive and periodic maintenance

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 algorithms, 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:
1. To formulate linear programming problem and solve using graphical method.
2. To solve LPP using simplex method
3. To formulate and solve transportation, assignment problems
4. To solve project management problems
5. To solve scheduling problems

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.

OUTCOMES:

Students will be able to:
1. Understand the costing concepts and their role in decision making
2. Understand the project management concepts and their various aspects in selection
3. Interpret costing concepts with project execution
4. Gain knowledge of costing techniques in service sector and various budgetary control techniques
5. Become familiar with quantitative techniques in cost management

TOTAL: 45 PERIODS
REFERENCES:
2. Charles T. Horngren and George Foster, Advanced Management Accounting, 1988

OE5095 COMPOSITE MATERIALS

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

UNIT I INTRODUCTION

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:
1. Know the characteristics of composite materials and effect of reinforcement in composite materials.
2. Know the various reinforcements used in composite materials.
3. Understand the manufacturing processes of metal matrix composites.
4. Understand the manufacturing processes of polymer matrix composites.
5. Analyze the strength of composite materials.

REFERENCES:

OE5096 WASTE TO ENERGY

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

UNIT I INTRODUCTION TO EXTRACTION OF ENERGY FROM WASTE
Classification of Waste as Fuel – Agro Based, Forest Residue, Industrial Waste - MSW – Conversion Devices – Incinerators, Gasifiers, Digestors

UNIT II BIOMASS PYROLYSIS

UNIT III BIOMASS GASIFICATION

UNIT IV BIOMASS COMBUSTION

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:
1. Understand the various types of wastes from which energy can be generated
2. Gain knowledge on biomass pyrolysis process and its applications
3. Develop knowledge on various types of biomass gasifiers and their operations
4. Gain knowledge on biomass combustors and its applications on generating energy
5. Understand the principles of bio-energy systems and their features

REFERENCES:

AUDIT COURSES (AC)

AX5091 ENGLISH FOR RESEARCH PAPER WRITING

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
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

UNIT III TITLE WRITING SKILLS
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
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
Useful phrases, checking Plagiarism, how to ensure paper is as good as it could possibly be the
OUTCOMES
1. Understand that how to improve your writing skills and level of readability
2. Learn about what to write in each section
3. Understand the skills needed when writing a Title
4. Understand the skills needed when writing the Conclusion
5. Ensure the good quality of paper at very first-time submission

REFERENCES

AX5092 DISASTER MANAGEMENT

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

Disaster Risk: Concept and Elements, Disaster Risk Reduction, Global and National Disaster Risk Situation. Techniques of Risk Assessment, Global Co-Operation in Risk Assessment and Warning, People’s Participation in Risk Assessment. Strategies for Survival

TOTAL: 30 PERIODS

OUTCOMES
1. Ability to summarize basics of disaster
2. Ability to explain a critical understanding of key concepts in disaster risk reduction and humanitarian response.
3. Ability to illustrate disaster risk reduction and humanitarian response policy and practice from multiple perspectives.
4. Ability to describe an understanding of standards of humanitarian response and practical relevance in specific types of disasters and conflict situations.
5. Ability to develop the strengths and weaknesses of disaster management approaches.

REFERENCES

AX5093  SANSKRIT FOR TECHNICAL KNOWLEDGE

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

UNIT I  ALPHABETS
Alphabets in Sanskrit

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

UNIT III  ORDER AND ROOTS
Order - Introduction of roots

UNIT IV  SANSKRIT LITERATURE
Technical information about Sanskrit Literature

UNIT V  TECHNICAL CONCEPTS OF ENGINEERING
Technical concepts of Engineering-Electrical, Mechanical, Architecture, Mathematics
OUTCOMES
1. Understanding basic Sanskrit language.
2. Write sentences.
3. Know the order and roots of Sanskrit.
4. Know about technical information about Sanskrit literature.
5. Understand the technical concepts of Engineering.

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

AX5094

VALUE EDUCATION

OBJECTIVES
1. Understand value of education and self-development
2. Imbibe good values in students
3. Let they should know about the importance of character

UNIT I

UNIT II

UNIT III

UNIT IV

OUTCOMES
Students will be able to
2. Learn the importance of Human values.
3. Developing the overall personality.

Suggested reading
OBJECTIVES
1. Understand the premises informing the twin themes of liberty and freedom from a civil rights perspective.
2. 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 of nationhood in the early years of Indian nationalism.
3. To address the role of socialism in India after the commencement of the Bolshevik Revolution in 1917 and 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:

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:

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.

OUTCOMES
Students will be able to:
1. Discuss the growth of the demand for civil rights in India for the bulk of Indians before the arrival of Gandhi in Indian politics.
2. Discuss the intellectual origins of the framework of argument that informed the conceptualization of social reforms leading to revolution in India.
3. 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.

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

AX5096 PEDAGOGY STUDIES

OBJECTIVES
1. Review existing evidence on their view topic to inform programme design and policy
2. Making undertaken by the DFLD, other agencies and researchers.
3. 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.

OUTCOMES
Students will be able to understand:
1. What pedagogical practices are being used by teachers informal and informal classrooms in developing countries?
2. What is the evidence on the effectiveness of these pedagogical practices, in what conditions, and with what population of learners?
3. 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

OBJECTIVES
1. To achieve overall health of body and mind
2. 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 Yoga 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
1. To learn to achieve the highest goal happily
2. To become a person with stable mind, pleasing personality, and determination
3. 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 (dons’t) - Verses- 71,73,75,78 (dos)

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 - Personalty of Role Model - shrimadbhagwadgeeta - 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
1. Study of Shrimad-Bhagwad-Geeta will help the student in developing his personality and achieve the highest goal in life
2. The person who has studied Geeta will lead the nation and mankind to peace and prosperity
3. Study of Neet is hatakam will help in developing versatile personality of students.

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

BRIDGE COURSES (BC)

MA5105 MATHEMATICAL FOUNDATIONS OF COMPUTER SCIENCE

OBJECTIVES:
• To introduce Mathematical Logic and their rules for validating arguments and programmes.
• To introduce counting principles for solving combinatorial problems.
• To give exposure to Graph models and their utility in connectivity problems.
• To introduce abstract notion of Algebraic structures for studying cryptographic and its related areas.
• To introduce Boolean algebra as a special algebraic structure for understanding logical circuit problems.

UNIT I LOGIC AND PROOFS

UNIT II COMBINATORICS
Mathematical Induction – Strong Induction and Well Ordering – The Basics of Counting - The

UNIT III   GRAPHS
Graphs and Graph Models – Graph Terminology and Special Types of Graphs – Matrix Representation of Graphs and Graph Isomorphism – Connectivity – Euler and Hamilton Paths.

UNIT IV   ALGEBRAIC STRUCTURES

UNIT V   LATTICES AND BOOLEAN ALGEBRA

TOTAL: 45 PERIODS

OUTCOMES:
On completion of the module the student should be able to:
1. Apply Mathematical Logic to validate logical arguments and programmes.
2. Apply Combinatorial Counting principles to solve application problems.
3. Apply graph model and graph techniques for solving network other connectivity related problems.
4. Apply algebraic ideas in developing cryptograph techniques for solving network security problems.
5. Apply Boolean laws in developing and simplifying logical circuits.

REFERENCES:

FUNDAMENTALS OF COMPUTING

OBJECTIVES:
- To learn programming using a structured programming language.
- To provide C programming exposure.
- To introduce foundational concepts of computer programming to students of different branches of Engineering and Technology.

UNIT I   INTRODUCTION
UNIT II  C PROGRAMMING BASICS  9

UNIT III  ARRAYS AND STRINGS  9

UNIT IV  POINTERS  9
Macros - Storage Classes –Basic Concepts of Pointers– Pointer Arithmetic - Example Problems - Basic File Operations

UNIT V  FUNCTIONS AND USER DEFINED DATA TYPES  9

TOTAL: 45 PERIODS

OUTCOMES:
On completion of the course, the students should be able to:
1. Write C program for simple applications
2. Formulate algorithm for simple problems
3. Analyze different data types and arrays
4. Perform simple search and sort.
5. Use Programming Language to solve problems.

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


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


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 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.

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

REFERENCES:

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 9
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.

TOTAL: 45 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:
OBJECTIVES:
- To study the fundamental concepts of Data Structures.
- To learn the concepts of Stacks, Queues, and its applications.
- To understand about Linked Lists.
- To understand the concepts of non-linear Data Structures.
- To learn the usage of sorting techniques.

UNIT I INTRODUCTION
Data Types – Abstract Data Types (ADTs) – Algorithm and Problem Solving – Data Structure: Array - Data Structure Operations - Algorithm: Complexity – Time, Space Tradeoff.

Suggested Activities:
- Flipped classroom on fundamentals of Data Structures.
- External learning – ADT’s, Arrays.
- Practical - Implementation of basic operations of Data Structures with Arrays.

Suggested Evaluation Methods:
- Quizzes on basic concepts of Data Structures.
- Assignments on basic operations in Arrays.
- Demonstration for practical learning implementations.

UNIT II STACK & QUEUE

Suggested Activities:
- Flipped classroom on basics of Stacks and Queues.
- External learning - Double ended queue.
- Practical - Implementation of Tower of Hanoi.
- Practical - Implementation of the Evaluation of expression using Stack.
- Practical - Implementation of any one application of Queue.

Suggested Evaluation Methods:
- Quizzes on Applications of Stack and Queue.
- Assignments on Double ended queues.
- Demonstration of the practical implementations.

UNIT III LINKED LISTS

Suggested Activities:
- Flipped classroom on basics of Linked Lists.
- External learning - Cursor based implementation of Linked Lists, applications of lists, double ended queue.
- Practical - Implementation of Polynomial using Lists.

Suggested Evaluation Methods:
- Quizzes on Linked Lists.
- Assignments on applications of Linked Lists.
- Demonstration of the practical implementations.

UNIT IV TREES

9

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 SORTING
Sorting Algorithms: Insertion Sort, Shell Sort, Quick Sort, Merge Sort.

Suggested Activities:
- Flipped classroom on different sorting techniques such as Bubble Sort, Selection Sort etc.
- External learning - Search algorithms, priority queues, external sorting, replacement selection 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 sorting.
- Assignments on creation and manipulation of 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 concepts of Data Structures.
2. Implement stack and queue Data Structure for an application.
3. Implement list Data Structure for an application.
4. Design and implement tree Data Structures.
5. Apply sorting algorithms for a given problem.
6. Choose appropriate Data Structure and implement a given application.

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:
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 I 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 that depicts 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 9

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 9

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

Suggested Evaluation Methods:
- Assignment on preparation of Software Configuration Management template for a software project.
- Calculation of Test metrics for Sample application.

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.

TOTAL: 45 PERIODS

REFERENCES:

OBJECTIVES:
- To understand the object-oriented concepts of Java.
- To learn GUI based application development using Java
- To learn the Threading concepts and generic collections used in Java.
- To learn network programming in Java.
- To design and develop small applications using core Java.

UNIT I
JAVA BASICS
Overview of Java – Java Fundamentals: Classes, Objects, Methods and Strings – Control Statements – Arrays and Array Lists.

Suggested Activities:
- Flipped classrooms on basics of Java.
- Learning and Implementation in the following topics.
  - Creation and manipulation of java programs using classes and objects.
  - Creation of simple exercises using Arrays and Array Lists.

Suggested Evaluation Methods:
- Quiz on Java Fundamentals.
- Demonstration of Java Programs using Classes and Objects.

UNIT II
CLASSES AND OBJECTS

Suggested Activities:
- Flipped classrooms on Exception Handling.
- Learning and Implementation in the following topics.
  - Creation and Manipulation of Java Programs using String Objects of Class String, String Builder and Characters.
  - Creation of Java applications using Interfaces, Packages and User Defined Exceptions.

Suggested Evaluation Methods:
- Quiz on Exception Handling.
- Demonstration of Java programs using Interfaces and Packages.

UNIT III
AWT AND THREADS

Suggested Activities:
- Flipped classrooms on Graphics Object and 2D API’s.
Learning and implementation in the following topics.
  o Creation of Java programs using Multithreading.
  o Creation of Java Applications using Applets and Swing.

Suggested Evaluation Methods:
  • Quiz on Java Graphical objects.
  • Demonstration and Evaluation of Java programs using AWT and Event Handling.

UNIT IV  IO AND GENERIC CLASSES
9
Files and Streams - Object Serialization – Recursion Concepts - Generic Collections Overview –
Collection Methods- Generic Classes-Implementation of Generic Methods– Overloading Generic
Methods -

Suggested Activities:
  • Flipped classroom on Generic Collections
  • Learning and implementation in the following topics.
    o Creation of Java Applications using Generics.
    o Creation of Java Applications using Files, Streams and Serialization.

Suggested Evaluation Methods:
  • Quiz on Java Generic Collections and Classes.
  • Demonstration and Evaluation of Java applications using File, Streams and Generics.

UNIT V  JAVA NETWORKING
9
Networking Overview - Manipulating URLs – Reading web pages – Client/Server Interaction using
Stream Sockets Connections – Connectionless Client/Server Interaction– Multicasting – Case
Study.

Suggested Activities:
  • Flipped classroom on basic networking and URL classes.
  • Implementation of Java programs using URL classes to read a live web page.
  • Creation of Chat applications using TCP and UDP protocols.
  • Creation of Group Chat application.

Suggested Evaluation Methods:
  • Quiz on Java Networking and URL classes.
  • Demonstration and Evaluation of Java applications using URL classes, sockets, datagrams
    and multicasting.

TOTAL: 45 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
  1. Implement object-oriented concepts of Java programming.
  2. Work with Interfaces, Packages and String Building concepts of Java.
  3. Design and implement java applications with graphical user interfaces.
  4. Design and implement java applications with appropriate input/output streams.
  5. Work with Networking concepts in Java.
  6. Design and develop real time applications using core Java.

REFERENCES:
**OBJECTIVES:**
- To introduce the concepts of structured programming language.
- To learn and implement linear data structures.
- To study and implement nonlinear data structures.
- To develop skills in design and implementation of data structures and their applications.
- To study and analyze the different sorting 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.
3. Implementation of simple programs in C using functions.
4. Implementation of simple programs in C using structures and unions.
5. Implementation of simple programs in C using pointers.
6. Implementation of stack using arrays and applications of stack.
8. Implementation of singly linked list.
9. Implementation of doubly linked list.
10. Implementation of circular linked list and applications of lists.
11. Implementation of binary search tree.

**TOTAL: 60 PERIODS**

**OUTCOMES:**
On completion of the course, the students will be able to:
1. Implement C programming concepts.
2. Develop simple C programs to solve an application.
3. Choose and apply linear data structure for a given application.
4. Choose and apply non-linear data structures for a given application.
5. Use sorting techniques for a given real world application.
6. Apply knowledge to solve computer science and information technology problems using the basics of C programming and the concepts of data structures.

**LABORATORY EXERCISES:**
- Create a database table, add constraints (primary key, unique, check, not null), insert rows, update, and delete rows using SQL DDL and DML commands.
- Create set of tables, add foreign key constraints, and incorporate referential integrity.
- Query the database tables using different ‘where’ clause conditions and implement aggregate functions.
- Query the database tables and explore sub queries and simple join operations.
- Query the database tables and explore natural, equi and outer joins.
- Write user defined functions and stored procedures in SQL.
- Execute complex transactions and realize DCL and TCL commands.
- Write SQL Triggers for insert, delete, and update operations in database table.
- Create a View and index for database tables with large number of records.
- Create an XML database and validate it using XML schema.
- Create Document, column and graph-based data using NOSQL database tools.
- 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.

**OBJECTIVES:**
- Understand Object Oriented features of Java.
- Learn about the GUI creation and Event Handling in Java.
- Understand the File concepts and generic collections of Java.
- Learn about Socket programming and networking in Java.

**EXPERIMENTS:**
1. Write a Java program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num 2 is displayed in the result field when the divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a Number Format Exception. If Num2 were Zero, the program would throw an Arithmetic Exception. Display the exception in a message dialog box.
2. Write a Java program that implements a multi-thread application that has three threads. First thread generates random integer every 1 second and if the value is even, second thread computes the square of the number and prints. If the value is odd, the third thread will print the value of cube of the number.
3. Write a Java program that simulates a traffic light. The program lets the user select one of three lights: red, yellow, or green with radio buttons. On selecting a button, an appropriate message with “Stop” or “Ready” or “Go” should appear above the buttons in selected color. Initially, there is no message shown.
4. Write a Java program that handles all mouse events and shows the event name at the center of the window when a mouse event is fired (Use Adapter classes).
5. Write a Java program for the following: Create a doubly linked list of elements. Delete a given element from the above list. Display the contents of the list after deletion.
6. Write a java program that implements Array Index out of bound Exception using built-in Exception.
7. Write a Java program that implements bank transactions using user denied exception.
8. Write a Java program to identify the significance of finally block in handling exceptions.
9. Write a Java program to generate multiple threads of creating clock pulses. (using runnable interface)
10. Write a Java program to implement mouse events like mouse pressed, mouse released, and mouse moved by means of adapter classes.
11. Write a Java program to demonstrate Window events on frame
12. Write a Java program to design the page authenticating username and password by using SWING.
13. Write a Java program to design a calculator by using Grid Layout.
14. Write a Java program that implements a simple client server application. The client sends data to server. The server receives the data uses it to produce a result and then sends the result back to the client then the client displays the result on the console.
15. Write a Java program that loads names and phone numbers from a text file where the data is organized as one line per record and each field in a record are separated by a tab (\t). It takes a name or phone number as input and prints the corresponding other value from the hash table (hint: use hash tables).

TOTAL: 60 PERIODS

OUTCOMES:
On completion of the course, the students will be able to:
1. Understand the structure and model of the Java programming language
2. Implement programs using the Object-Oriented features of Java
3. Use the Java programming language for various programming technologies
4. Develop software in the Java programming language,
5. Evaluate user requirements for software functionality required to decide whether the Java programming language can meet user requirements
6. Propose the use of certain technologies by implementing them in the Java programming language to solve the given problem