Bansilal Ramnath Agarwal Charitable Trust’s
Vishwakarma Institute of Technology
(An Autonomous Institute affiliated to University of Pune)

Structure & Syllabus of
B.Tech. (Computer Engineering)

Pattern ‘F-11’
Effective from Academic Year 2014-15

Prepared by: - Board of Studies in Computer Engineering
Approved by: - Academic Board, Vishwakarma Institute of Technology, Pune

Signed by

Chairman – BOS       Chairman – Academic Board

Content

<table>
<thead>
<tr>
<th>Sr. No.</th>
<th>Title</th>
<th>Page No.</th>
</tr>
</thead>
</table>

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><strong>Program Educational Objectives and Program Outcome of B.Tech</strong></td>
<td>8</td>
</tr>
<tr>
<td></td>
<td><em>(Computer Engineering)</em></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td><strong>Course Structure - Module I and II</strong></td>
<td>11</td>
</tr>
<tr>
<td>2.1</td>
<td>CS10102 Computer programming <em>(Theory Course)</em></td>
<td>18</td>
</tr>
<tr>
<td>2.2</td>
<td>CS10302 Computer Programming <em>(Laboratory)</em></td>
<td>21</td>
</tr>
<tr>
<td>3</td>
<td><strong>Course Structure - Module III</strong></td>
<td>23</td>
</tr>
<tr>
<td>4</td>
<td><strong>Course Syllabi for Courses - Module III</strong></td>
<td>24</td>
</tr>
<tr>
<td>4.1</td>
<td>CS20115 Discrete Structures <em>(Theory Course)</em></td>
<td>25</td>
</tr>
<tr>
<td>4.2</td>
<td>CS20111 Data Structures <em>(Theory Course)</em></td>
<td>28</td>
</tr>
<tr>
<td>4.3</td>
<td>CS20108 Computer Organization <em>(Theory Course)</em></td>
<td>30</td>
</tr>
<tr>
<td>4.4</td>
<td>CS21109 Applied Electronics <em>(Theory Course)</em></td>
<td>32</td>
</tr>
<tr>
<td>4.5</td>
<td>CS20113 Digital Electronics and Logic Design <em>(Theory Course)</em></td>
<td>35</td>
</tr>
<tr>
<td>4.6</td>
<td>CS20115 Discrete Structures <em>(Tutorial)</em></td>
<td>38</td>
</tr>
<tr>
<td>4.7</td>
<td>CS20113 Digital Electronics and Logic Design <em>(Tutorial)</em></td>
<td>40</td>
</tr>
<tr>
<td>4.8</td>
<td>CS20311 Data structures <em>(Laboratory Course)</em></td>
<td>41</td>
</tr>
<tr>
<td>4.9</td>
<td>CS20313 Digital Electronics <em>(Laboratory Course)</em></td>
<td>42</td>
</tr>
<tr>
<td>4.10</td>
<td>CS27401 Mini Project</td>
<td>44</td>
</tr>
<tr>
<td>4.11</td>
<td>CS24301 VB.NET OR CS24303 C#.NET <em>(Skill Development Course)</em></td>
<td>45</td>
</tr>
<tr>
<td>4.12</td>
<td>Elective –Soft Skills</td>
<td></td>
</tr>
<tr>
<td>4.13</td>
<td>CS20401 Comprehensive Viva Voce</td>
<td></td>
</tr>
<tr>
<td>4.14</td>
<td>Institute Elective</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td><strong>Course Structure - Module IV</strong></td>
<td>49</td>
</tr>
<tr>
<td>6</td>
<td><strong>Course Syllabi for Courses - Module IV</strong></td>
<td>50</td>
</tr>
<tr>
<td>6.1</td>
<td>CS21104 Mathematical Transformations and Applications <em>(Theory Course)</em></td>
<td>52</td>
</tr>
<tr>
<td>6.2</td>
<td>CS20110 Computer Graphics <em>(Theory Course)</em></td>
<td>55</td>
</tr>
<tr>
<td>6.3</td>
<td>CS20105 Principles of Programming Languages <em>(Theory Course)</em></td>
<td>58</td>
</tr>
<tr>
<td>6.4</td>
<td>CS21112 Data Communication <em>(Theory Course)</em></td>
<td>61</td>
</tr>
</tbody>
</table>
### 6.5 CS20114 Microprocessor and Interfacing (Theory Course) 64

### 6.6 CS20114 Microprocessor and Interfacing (Tutorial) 67

### 6.7 CS21104 Mathematical Transforms and Applications (Tutorial) 69

### 6.8 CS20314 Microprocessor and Interfacing (Laboratory Course) 71

### 6.9 CS20305 Principles of Programming Languages (Laboratory Course) 74

### 6.10 CS20310 Computer Graphics (Laboratory Course) 76

### 6.11 CS27402 Mini Project 77

### 6.12 CS24302 ASP.NET OR 78

### 6.13 CS24304 Python (Skill Development Course) 80

### 6.14 CS20402 Comprehensive Viva Voce -

### 7 Course Structure - Module V 84

### 8 Course Syllabi for Courses - Module V 85

### 8.1 CS30101 Operating Systems (Theory Course) 87

### 8.2 CS30116 Computer Networks (Theory Course) 90

### 8.3 CS30105 Theory of Computation (Theory Course) 93

### 8.4 CS31113 Microprocessors and Microcontrollers (Theory Course) 96

### 8.5 CS31115 Advanced Data Structures (Theory Course) 98

### 8.6 CS30101 Operating Systems (Tutorial) 100

### 8.7 CS31215 Advanced Data Structures (Tutorial) 101

### 8.8 CS30303 Operating Systems (Laboratory Course) 103

### 8.9 CS30316 Computer Networks (Laboratory Course) 105

### 8.10 CS31313 Microprocessors and Microcontrollers (Laboratory Course) 107

### 8.11 CS37401 Mini Project 108

### 8.12 Professional Development Course (Institute Level) -

### 8.13 CS30401 Comprehensive Viva Voce -

### 8.14 CS37301 Seminar 109

### 8.15 CS37302 Project Stage - 1 110
### Course Structure - Module VI

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Title</th>
<th>Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS30102</td>
<td>Software Engineering (Theory Course)</td>
<td></td>
</tr>
<tr>
<td>CS30106</td>
<td>Database Management Systems (Theory Course)</td>
<td></td>
</tr>
<tr>
<td>CS30108</td>
<td>Design and Analysis of Algorithm (Theory Course)</td>
<td></td>
</tr>
<tr>
<td>CS30114</td>
<td>Systems Programming (Theory Course)</td>
<td></td>
</tr>
<tr>
<td>CS31119</td>
<td>Object Oriented Modeling and Design (Theory Course)</td>
<td></td>
</tr>
<tr>
<td>CS30102</td>
<td>Software Engineering (Tutorial)</td>
<td></td>
</tr>
<tr>
<td>CS30108</td>
<td>Design and Analysis of Algorithm (Tutorial)</td>
<td></td>
</tr>
<tr>
<td>CS30314</td>
<td>System programming (Laboratory Course)</td>
<td></td>
</tr>
<tr>
<td>CS31319</td>
<td>Object Oriented Modeling and Design (Laboratory Course)</td>
<td></td>
</tr>
<tr>
<td>CS30306</td>
<td>Database Management Systems (Laboratory Course)</td>
<td></td>
</tr>
<tr>
<td>CS37402</td>
<td>Mini Project</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Professional Development Course (Institute Level)</td>
<td></td>
</tr>
<tr>
<td>CS30402</td>
<td>Comprehensive Viva Voce</td>
<td></td>
</tr>
<tr>
<td>CS37301</td>
<td>Seminar</td>
<td></td>
</tr>
</tbody>
</table>

### Course Structure - Module VII

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Title</th>
<th>Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS40115</td>
<td>Distributed Computing (Theory Course)</td>
<td></td>
</tr>
<tr>
<td>CS40114</td>
<td>Business Intelligence and Analytics (Theory Course)</td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Elective Group I (Theory Course)</strong></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Title</th>
<th>Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS42105</td>
<td>Mobile Computing</td>
<td></td>
</tr>
<tr>
<td>CS42131</td>
<td>Enterprise Systems</td>
<td></td>
</tr>
<tr>
<td>CS42127</td>
<td>Cloud Computing</td>
<td></td>
</tr>
<tr>
<td>CS42103</td>
<td>Parallel Computing on GPU</td>
<td></td>
</tr>
<tr>
<td>CS42132</td>
<td>Network Security</td>
<td></td>
</tr>
<tr>
<td>CS42101</td>
<td>Advanced Computer Graphics</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Title</th>
<th>Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS42125</td>
<td>Randomized and Approximation Algorithms</td>
<td></td>
</tr>
<tr>
<td>CS42113</td>
<td>Digital Signal Processing</td>
<td></td>
</tr>
<tr>
<td>CS42119</td>
<td>Information Retrieval</td>
<td></td>
</tr>
<tr>
<td>Course Code</td>
<td>Course Name</td>
<td>Credits</td>
</tr>
<tr>
<td>------------</td>
<td>-------------------------------------------------</td>
<td>---------</td>
</tr>
<tr>
<td>CS42114</td>
<td>Product Design</td>
<td>176</td>
</tr>
<tr>
<td>CS42115</td>
<td>Human Computer Interaction</td>
<td>179</td>
</tr>
<tr>
<td>CS42134</td>
<td>Modeling and Simulation</td>
<td>182</td>
</tr>
<tr>
<td>CS40115</td>
<td>Distributed Computing (Tutorial Course)</td>
<td>185</td>
</tr>
<tr>
<td>CS42125</td>
<td>Randomized and Approximation Algorithms</td>
<td>187</td>
</tr>
<tr>
<td>CS42113</td>
<td>Digital Signal Processing</td>
<td>189</td>
</tr>
<tr>
<td>CS42119</td>
<td>Information Retrieval</td>
<td>191</td>
</tr>
<tr>
<td>CS42114</td>
<td>Product Design</td>
<td>192</td>
</tr>
<tr>
<td>CS42115</td>
<td>Human Computer Interaction</td>
<td>193</td>
</tr>
<tr>
<td>CS42134</td>
<td>Modeling and Simulation</td>
<td>194</td>
</tr>
<tr>
<td>CS40314</td>
<td>Business Intelligence and Analytics (Laboratory Course)</td>
<td>196</td>
</tr>
<tr>
<td>CS42305</td>
<td>Mobile Computing</td>
<td>198</td>
</tr>
<tr>
<td>CS42331</td>
<td>Enterprise Systems</td>
<td>200</td>
</tr>
<tr>
<td>CS42327</td>
<td>Cloud Computing</td>
<td>202</td>
</tr>
<tr>
<td>CS42303</td>
<td>Parallel Computing on GPU</td>
<td>203</td>
</tr>
<tr>
<td>CS42332</td>
<td>Network Security</td>
<td>204</td>
</tr>
<tr>
<td>CS42301</td>
<td>Advanced Computer Graphics</td>
<td>206</td>
</tr>
<tr>
<td>CS447303</td>
<td>Project Stage - II</td>
<td>208</td>
</tr>
</tbody>
</table>

13. **Course Structure - Module VIII**

14. **Course Syllabi for Courses - Module VIII**

14.1 CS40106 Compiler Design (Theory Course)

14.2 CS40110 Artificial Intelligence (Theory Course)

14.3 **Elective Group III (Theory Course)**

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Name</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS42120</td>
<td>Data Mining</td>
<td>218</td>
</tr>
<tr>
<td>CS42128</td>
<td>Machine Learning</td>
<td>221</td>
</tr>
<tr>
<td>CS42104</td>
<td>Neural Networks</td>
<td>224</td>
</tr>
<tr>
<td>CS42111</td>
<td>Software testing and Quality Assurance</td>
<td>227</td>
</tr>
<tr>
<td>CS42124</td>
<td>Algorithmic Number Theory and Algebra</td>
<td>230</td>
</tr>
</tbody>
</table>

14.4 **Elective Group IV (Theory Course)**

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Name</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS42118</td>
<td>Geographical Information Systems</td>
<td>234</td>
</tr>
<tr>
<td>CS42129</td>
<td>Management Information Systems</td>
<td>237</td>
</tr>
<tr>
<td>CS42121</td>
<td>Advanced Computer Architecture</td>
<td>240</td>
</tr>
<tr>
<td>CS42116</td>
<td>Convergence Technologies</td>
<td>243</td>
</tr>
<tr>
<td>Course Code</td>
<td>Course Title</td>
<td>Credits</td>
</tr>
<tr>
<td>-------------</td>
<td>----------------------------------------------------------------</td>
<td>---------</td>
</tr>
<tr>
<td>CS42130</td>
<td>Embedded Systems</td>
<td>246</td>
</tr>
<tr>
<td>CS42133</td>
<td>Image Processing</td>
<td>248</td>
</tr>
<tr>
<td>CS42120</td>
<td>Data Mining</td>
<td>250</td>
</tr>
<tr>
<td>CS42128</td>
<td>Machine Learning</td>
<td>251</td>
</tr>
<tr>
<td>CS42104</td>
<td>Neural Networks</td>
<td>253</td>
</tr>
<tr>
<td>CS42111</td>
<td>Software testing and Quality Assurance</td>
<td>255</td>
</tr>
<tr>
<td>CS42124</td>
<td>Algorithmic Number Theory and Algebra</td>
<td>257</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Elective Group III (Tutorial)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS42118</td>
</tr>
<tr>
<td>CS42129</td>
</tr>
<tr>
<td>CS42121</td>
</tr>
<tr>
<td>CS42116</td>
</tr>
<tr>
<td>CS42130</td>
</tr>
<tr>
<td>CS42133</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Elective Group IV (Tutorial)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS4118</td>
</tr>
<tr>
<td>CS42129</td>
</tr>
<tr>
<td>CS42121</td>
</tr>
<tr>
<td>CS42116</td>
</tr>
<tr>
<td>CS42130</td>
</tr>
<tr>
<td>CS42133</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Course Syllabi for PD Courses in TY B.Tech (Computer Engineering)</th>
</tr>
</thead>
<tbody>
<tr>
<td>15.1 CS33303</td>
</tr>
<tr>
<td>15.2 CS33312</td>
</tr>
<tr>
<td>15.3 CS33313</td>
</tr>
<tr>
<td>15.4 CS33306</td>
</tr>
<tr>
<td>15.5 CS33310</td>
</tr>
<tr>
<td>15.6 CS33314</td>
</tr>
<tr>
<td>15.7 CS33311</td>
</tr>
<tr>
<td>15.8 CS33315</td>
</tr>
<tr>
<td>15.9 CS33307</td>
</tr>
</tbody>
</table>
Program Educational Objectives (PEO)
B.Tech (Computer Engineering)
List of Programme Education Objectives [PEO] and Programme Outcomes [PO]

<table>
<thead>
<tr>
<th>PEO</th>
<th>PEO Focus</th>
<th>PEO Statement</th>
</tr>
</thead>
<tbody>
<tr>
<td>PEO1</td>
<td>Preparation</td>
<td>To prepare the students as a committed technology workforce by providing them global educational platform with innovative practices resulting in computing artifacts realization</td>
</tr>
<tr>
<td>PEO2</td>
<td>Core competence</td>
<td>To impart adequate mathematical and computing theory knowledge basis leading to sustainable computer engineering solutions development</td>
</tr>
<tr>
<td>PEO3</td>
<td>Breadth</td>
<td>To inculcate problem solving skills and engineering practices in students adhering to well-formed technical specifications and constraints with the help of sound methods, tools and techniques</td>
</tr>
<tr>
<td>PEO4</td>
<td>Professionalism</td>
<td>To instill in the students professional and ethical practices by following effective guidelines to acquire aptitude, attitude and desire beneficial in societal context</td>
</tr>
<tr>
<td>PEO5</td>
<td>Learning Environment</td>
<td>To promote aspiring students for continuing education, engineering certifications and entrepreneurship in emerging areas of computing</td>
</tr>
</tbody>
</table>
List of Programme Outcomes [PO]

Graduates will be able

<table>
<thead>
<tr>
<th>PO</th>
<th>Graduate Attributes</th>
<th>PO Statement</th>
</tr>
</thead>
<tbody>
<tr>
<td>P01</td>
<td>GA: 1 Engineering Knowledge</td>
<td>1. To apply scientific, mathematical and computing fundamentals in order to devise engineering solution for real world problems.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2. To apply computer theory and algorithmic principles to innovatively craft solutions by context and development.</td>
</tr>
<tr>
<td>P02</td>
<td>GA: 2 Problem Analysis</td>
<td>3. To discover and infer computing problem situations, resulting in physical model, mathematical model or graphical model depicting the overall problem.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>4. To systematize functional specifications of target computing environment by adequate consideration of technology infrastructure, boundary conditions and constraints.</td>
</tr>
<tr>
<td>P03</td>
<td>GA: 3: Design/Development of solution</td>
<td>5. To conceive well-formed design specifications and constructs demonstrating correct compositional system structure with implementation-centric considerations.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>6. To incorporate architectural styles and design patterns to assimilate new facts, information and ideas about the design.</td>
</tr>
<tr>
<td>P04</td>
<td>GA: 4: Conduct Investigation of Complex Problem</td>
<td>7. To interpret reference data and program pragmatics for analyzable experimental results derivation.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>8. To judge and relate complexity issues and levels by making use of standardized verification and validation techniques.</td>
</tr>
<tr>
<td>P05</td>
<td>GA: 5: Modern Tool Usage</td>
<td>9. To operationalize and utilize the state-of-the-art CASE tools for engineering artifacts construction.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>10. To correlate and hypothesize problems for recognizing new or unfamiliar problem patterns.</td>
</tr>
<tr>
<td>P06</td>
<td>GA: 6: The Engineer and Society</td>
<td>11. To minimize adverse effects on the environment for their own and succeeding generations by respecting published facts and guidelines.</td>
</tr>
<tr>
<td>P07</td>
<td>GA: 7: Environment and sustainability</td>
<td>12. To consider the impact and benefits of engineering achievements in exploitation and management of technology on environment and society.</td>
</tr>
<tr>
<td>P08</td>
<td>GA: 8: Ethics</td>
<td>13. To prepare and present engineering evidence, theory and interpretations honestly, accurately and without bias.</td>
</tr>
<tr>
<td>P09</td>
<td>GA: 9: Individual and Team Work</td>
<td>14. To demonstrate high standards of professional conduct, openness and fairness by maintaining due respect towards rights and reputation of team members and development organization.</td>
</tr>
<tr>
<td>P10</td>
<td>GA: 10: Communication</td>
<td>15. To demonstrate deep listening, learning, leadership and managerial skills to solve complex engineering problems in teams.</td>
</tr>
<tr>
<td>P11</td>
<td>GA: 11: Lifelong Learning</td>
<td>16. To become part of a valuable body of knowledge in competitive computing areas.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>17. To acquire responsible positions in government, industry and society by</td>
</tr>
</tbody>
</table>
continuously learning and researching.

| PO12 | GA: 12: Project Management and Finance | 18. To creatively devise and incorporate project-specific processes supported by rigorous standards applicable to professional engineering bodies. |
## Module 1

<table>
<thead>
<tr>
<th>Code</th>
<th>Subject</th>
<th>Type</th>
<th>Teaching Scheme</th>
<th>Assessment Scheme</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>L</td>
<td>P</td>
<td>Tut.</td>
</tr>
<tr>
<td>HS10109</td>
<td>Linear Algebra and Random Variables</td>
<td>Theory – Core</td>
<td>3</td>
<td>-</td>
<td>1</td>
</tr>
<tr>
<td>HS10103</td>
<td>Modern Physics</td>
<td>Theory – Core</td>
<td>3</td>
<td>-</td>
<td>1</td>
</tr>
<tr>
<td>CH10101</td>
<td>Chemistry</td>
<td>Theory – Core</td>
<td>3</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>ME10101</td>
<td>Engineering Graphics</td>
<td>Theory – Core</td>
<td>3</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>HS16101</td>
<td>Sociology</td>
<td>Theory – OE</td>
<td>2</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>HS16103</td>
<td>Psychology</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HS16105</td>
<td>Philosophy</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HS16107</td>
<td>EVS</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HS10301</td>
<td>Engineering Graphics Lab</td>
<td>Lab – Core</td>
<td>-</td>
<td>2</td>
<td>-</td>
</tr>
<tr>
<td>CH10301</td>
<td>Science Lab</td>
<td>Lab – Core</td>
<td>-</td>
<td>2</td>
<td>-</td>
</tr>
<tr>
<td>HS17401</td>
<td>Mini Project</td>
<td>Project</td>
<td>-</td>
<td>4</td>
<td>-</td>
</tr>
<tr>
<td><strong>TOTAL</strong></td>
<td></td>
<td></td>
<td><strong>14</strong></td>
<td><strong>8</strong></td>
<td><strong>2</strong></td>
</tr>
</tbody>
</table>

## Module 2

<table>
<thead>
<tr>
<th>Code</th>
<th>Subject</th>
<th>Type</th>
<th>Teaching Scheme</th>
<th>Assessment Scheme</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>L   P  Tut.</td>
<td>ISA  Test 1  Test 2  HA  Tut.  CA  ESA</td>
<td></td>
</tr>
<tr>
<td>HS10102</td>
<td>Differential Integral and Calculus</td>
<td>Theory - Core</td>
<td>3   -  1</td>
<td>10  20  5  5 - -</td>
<td>60 4</td>
</tr>
<tr>
<td>HS10104</td>
<td>Engineering Mechanics</td>
<td>Theory – Core</td>
<td>3   -  1</td>
<td>10  20  5  5 -</td>
<td>60 4</td>
</tr>
<tr>
<td>HS10108</td>
<td>Electrical Engineering Fundamentals</td>
<td>Theory – Core</td>
<td>3   - -</td>
<td>15  20  5 - -</td>
<td>60 3</td>
</tr>
<tr>
<td>CS10102</td>
<td>Computer Programming</td>
<td>Theory – Core</td>
<td>3   - -</td>
<td>15  20  5 - -</td>
<td>60 3</td>
</tr>
<tr>
<td>HS16102</td>
<td>Economics Management</td>
<td>Theory – OE</td>
<td>2   - -</td>
<td>15  20  5 - -</td>
<td>60 2</td>
</tr>
<tr>
<td>HS16104</td>
<td>Technology Cost &amp; Acc.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HS16106</td>
<td>Business Law</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CS10302</td>
<td>Computer Programming</td>
<td>Lab – Core</td>
<td>-   2 - -</td>
<td>- - - -</td>
<td>70 30 1</td>
</tr>
<tr>
<td>HS10306</td>
<td>Engineering Lab</td>
<td>Lab – Core</td>
<td>-   2 - -</td>
<td>- - - -</td>
<td>70 30 1</td>
</tr>
<tr>
<td>HS17402</td>
<td>Mini Project</td>
<td>Project</td>
<td>-   4 - -</td>
<td>- - - -</td>
<td>70 30 2</td>
</tr>
<tr>
<td><strong>TOTAL</strong></td>
<td></td>
<td></td>
<td>14  08 2</td>
<td>2</td>
<td>20</td>
</tr>
</tbody>
</table>
## F.Y. B. Tech. Structure with effect from Academic Year 2015-16

### Semester I – Irrespective of Module

<table>
<thead>
<tr>
<th>Code</th>
<th>Subject</th>
<th>Type</th>
<th>Teaching Scheme</th>
<th>Assessment Scheme</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>L   P   Tut.</td>
<td>ISA</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Test 1</td>
<td>Test 2</td>
<td>HA</td>
</tr>
<tr>
<td>HS10107</td>
<td>Communication Skill</td>
<td>Comm. Skill</td>
<td>-   2</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>HS153xx</td>
<td>General Proficiency</td>
<td>GP</td>
<td>-   2</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>HS15301</td>
<td>English I</td>
<td>Language</td>
<td>-   2</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>HS15302</td>
<td>French I</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HS15303</td>
<td>German I</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HS15304</td>
<td>Spanish I</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HS15305</td>
<td>Japanese I</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HS14301</td>
<td>Engineering Workshop</td>
<td>Workshop</td>
<td>-   2</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

**TOTAL** | -   8 | - |     |     |       |       |    |     | 4
### F.Y. B. Tech. Structure with effect from Academic Year 2015-16

#### Semester II – Irrespective of Module

<table>
<thead>
<tr>
<th>Code</th>
<th>Subject</th>
<th>Type</th>
<th>Teaching Scheme</th>
<th>Assessment Scheme</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>L</td>
<td>P</td>
<td>Tut.</td>
</tr>
<tr>
<td>HS17301</td>
<td>General seminar I</td>
<td>Comm. Skill</td>
<td>-</td>
<td>2</td>
<td>-</td>
</tr>
<tr>
<td>HS153xx</td>
<td>General Proficiency</td>
<td>GP</td>
<td>-</td>
<td>2</td>
<td>-</td>
</tr>
<tr>
<td>HS15306</td>
<td>French II</td>
<td>Language</td>
<td>-</td>
<td>2</td>
<td>-</td>
</tr>
<tr>
<td>HS15307</td>
<td>German II</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HS15308</td>
<td>Spanish II</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HS15309</td>
<td>Japanese II</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HS15310</td>
<td>English II</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HS14302</td>
<td>Trade Workshop</td>
<td>Workshop</td>
<td>-</td>
<td>2</td>
<td>-</td>
</tr>
</tbody>
</table>

**TOTAL** - 8 - - - - - 4

**HS153xx : General Proficiency Courses as per following list**
## List of General Proficiency Courses

**FY B Tech**

**AY 2015-16**

<table>
<thead>
<tr>
<th>Sr. No.</th>
<th>Course Code</th>
<th>Name of Course</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>HS15311</td>
<td>Flute</td>
</tr>
<tr>
<td>2</td>
<td>HS15312</td>
<td>Guitar</td>
</tr>
<tr>
<td>3</td>
<td>HS15313</td>
<td>Tabla</td>
</tr>
<tr>
<td>4</td>
<td>HS15314</td>
<td>Bharat Natyam</td>
</tr>
<tr>
<td>5</td>
<td>HS15317</td>
<td>Yoga</td>
</tr>
<tr>
<td>6</td>
<td>HS15318</td>
<td>Pranayam</td>
</tr>
<tr>
<td>7</td>
<td>HS15319</td>
<td>Aerobics</td>
</tr>
<tr>
<td>8</td>
<td>HS15321</td>
<td>Photography</td>
</tr>
<tr>
<td>9</td>
<td>HS15322</td>
<td>Digital Photography</td>
</tr>
<tr>
<td>10</td>
<td>HS15323</td>
<td>Volleyball</td>
</tr>
<tr>
<td>11</td>
<td>HS15324</td>
<td>Chess</td>
</tr>
<tr>
<td>12</td>
<td>HS15325</td>
<td>Taekwondo</td>
</tr>
<tr>
<td>13</td>
<td>HS15326</td>
<td>Film Appreciation</td>
</tr>
<tr>
<td>14</td>
<td>HS15327</td>
<td>Shares and Stocks</td>
</tr>
<tr>
<td>15</td>
<td>HS15328</td>
<td>Fundamentals of Banking</td>
</tr>
<tr>
<td>16</td>
<td>HS15329</td>
<td>Nutrition and Fitness</td>
</tr>
<tr>
<td>17</td>
<td>HS15330</td>
<td>Spirit of Entrepreneurship</td>
</tr>
<tr>
<td>18</td>
<td>HS15331</td>
<td>Memory Techniques</td>
</tr>
<tr>
<td>19</td>
<td>HS15332</td>
<td>Sanskrit Sambhashanam</td>
</tr>
<tr>
<td>20</td>
<td>HS15333</td>
<td>Numerology</td>
</tr>
</tbody>
</table>
CS10102:: COMPUTER PROGRAMMING

Credits: 03

Teaching Scheme: - Theory 3 Hrs/Week

Prerequisites:

Unit 1 : Introduction to Programming: (8+2 Hrs)
PART A: Problem solving using computers and logic design. Algorithms and their representations: flowcharts, pseudo code. Designing algorithms for problems like finding min-max, mean, median, mode, mensuration and roots of a quadratic equation. Concept of programming languages for implementing algorithms – levels of languages. Role of assemblers, compilers, linker, loader, interpreter in program execution.


PART B: Bits and bytes – importance of digital representation in computers. Number System and algorithms for inter conversions. C programming on Linux and Windows.

Unit II : Flow of Control (8+2 Hrs)

Arrays: Concept, declaration and initialization of arrays, accessing individual elements of array. Use of arrays in sorting, searching. Concept of 2-D array (Matrix), row major and column major representation of array, address calculation for accessing the individual element.

PART B : Static variables and constants in C language.

Unit 3: Functions (8+2 Hrs)
PART A : Need of functions, function declaration, definition and call. Inbuilt functions and user defined functions. Passing arguments to a function, returning values from a function. Scope of variable, local and global variable. Access specifiers. Passing arrays to functions.

PART B: Preprocessor and preprocessor directives: macro substitution, difference between macro and functions.

Unit 4: Pointers and Strings (8+2 Hrs)

Strings: Strings as arrays, character array versus strings, reading strings, writing strings, user defined functions for string operations – copy, concatenate, length, reverse, converting case, appending, comparing two string, extracting a substring. Array of strings.

PART B: Const keyword in C, standard string library functions in string.h for string manipulation.

Unit 5: Structures and File Handling (8+2 Hrs)
PART A: Structures: Notion, declaration and initialization, structure variables, accessing and assigning values of the fields, "size of" operator, functions and structures, arrays of structures, nested structures, pointers and structures, passing structure to a function and returning structure from function. Dynamic memory allocation, type casting. Introduction to self referential structures, linked list as a dynamic alternative to arrays.

File Handling in C: file types, file opening modes, file handling I/O – fprintf, fscanf, fwrite, fread, fseek. File pointers. Implementing basic file operations in C.


Text Books

Reference Books

Additional Reading

Course Outcomes:
Upon completion of the course, graduates will be able to -
1. List procedural programming benefits to construct concise solutions
2. Interpret and develop naturo-visual representation of problem in hand.
3. Apply available algorithmic principles to general efficient solutions
4. Justify modular programming approach by making use of elementary as well as superior data structures.
5. Apply programming fundamentals with generic prototype.
6. Evaluate and manipulate given solutions in reengineered view
CS10302: COMPUTER PROGRAMMING LAB

Credits: 01

Teaching Scheme: - Laboratory 2 Hrs/Week

Prerequisites:

List of Practicals

1. Study of most important DOS/UNIX commands.

2. Write a program in C to find largest element / average of given N elements / sum / reverse of a given integer.

3. Write a program in C to implement a simple mathematical calculator

4. Write a program in C to read an integer and display each of the digits of an integer in English.

5. Write a C program to generate all the prime numbers between 1 and n, where n is a value supplied by the user.

6. Write a program in C to perform Addition / Subtraction / Multiplication of two Matrices. Also determine whether the matrix is symmetric / skewed.

7. Write a program in C to carry out following operations on strings using string library Functions:
   a. Length of a string.
   b. Copy a string.
   c. Concatenation of strings.

8. Write a program in C to carry out following operations on strings without using string library functions
   a. Compare two strings.
   b. Reverse given string.
   c. To check if the given string is a palindrome or not.

9. Write a program in C to carry out following operations on strings using pointers.
   a. Length of a string.
   b. Concatenation of strings.
   c. Copy of string
   d. Compare two strings.

10. Write a C program that works with complex numbers using a structure. Perform the following operations:
    a. Reading a complex number.
b. Addition of two complex numbers.
c. Writing a complex number.
d. Multiplication of two complex numbers.

11. Write a C program to create a database of students by using array of structure and perform following operations on it.
a. Accept/modify record of student
b. Search a particular record
c. Display all records

12. Write a program in C that use both recursive and non-recursive functions to find the Factorial / GCD (greatest common divisor) of two given integers / Fibonacci series.

13. Write a program in C to sort n integers using bubble / merge sort.

14. Write a program in C to search a number in a given list using linear / binary search.

Text Books

Reference Books

Additional Reading
MODULE VII
## Course Structure

<table>
<thead>
<tr>
<th>Code</th>
<th>Subject Name</th>
<th>Teaching Scheme (Hrs./week)</th>
<th>Examination Scheme</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Lectures</td>
<td>Tutorial</td>
<td>HA</td>
</tr>
<tr>
<td>CS40115</td>
<td>Distributed Computing</td>
<td>3</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>CS40114</td>
<td>Business Intelligence and Analytics</td>
<td>3</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>CS42101</td>
<td>Advanced Computer Graphics</td>
<td>3</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>CS42107</td>
<td>Cloud Computing</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CS42103</td>
<td>Parallel Computing on GPU</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CS42105</td>
<td>Mobile Computing</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CS42131</td>
<td>Enterprise Systems</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CS42132</td>
<td>Networks Security</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ELECTIVE I</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ELECTIVE II</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>TOTAL</td>
<td>14</td>
<td>2</td>
<td></td>
</tr>
</tbody>
</table>

### Laboratory Courses

<table>
<thead>
<tr>
<th>Code</th>
<th>Subject Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS42125</td>
<td>Randomized and Approximation Algorithms</td>
</tr>
<tr>
<td>CS42113</td>
<td>Digital Signal Processing</td>
</tr>
<tr>
<td>CS42119</td>
<td>Information Retrieval</td>
</tr>
<tr>
<td>CS42114</td>
<td>Product Design</td>
</tr>
<tr>
<td>CS42115</td>
<td>Human Computer Interaction</td>
</tr>
<tr>
<td>CS42134</td>
<td>Modeling and Simulation</td>
</tr>
<tr>
<td>Code</td>
<td>Subject Name</td>
</tr>
<tr>
<td>---------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>CS40314</td>
<td>Business Intelligence and Analytics</td>
</tr>
<tr>
<td></td>
<td>Elective I</td>
</tr>
<tr>
<td>CS47303</td>
<td>PROJECT STAGE 2 (B.Tech. Semester I)</td>
</tr>
<tr>
<td></td>
<td>Irrespective of Module</td>
</tr>
<tr>
<td></td>
<td>TOTAL</td>
</tr>
</tbody>
</table>

**Abbreviations Used**

- HA: Home Assignment
- TA: Teacher Assessment
- ISE: In Sem. Examination
- ESE: End Sem. Examination
- CA: Continuous Assessment
- *: Alternate Week Laboratory
- ^: Detail Enclosed Separately

**TOTAL CREDITS**: 20
CS40115:: DISTRIBUTED COMPUTING

Credits: 03  
Teaching Scheme: - Theory 3 Hrs/Week

Prerequisites: Operating Systems.

Unit 1: Introduction  
PART B: Conceptual design of a Distributed File System based on concepts learned in Unit I. Reading: Hadoop/MapReduce.

Unit 2: Communication  
PART B: Case studies of existing RPC implementation. Design a RPC/RMI mechanism based on the case studies and concepts learned in class.

Unit 3: Synchronization  
PART B: Comparative analysis of time synchronization/mutual exclusion/election algorithms implementations in well-established distributed systems.

Unit 4: Fault Tolerance  
PART B: Identify the issues that can arise in your DFS prototype in case of various failures. Based on the concepts learned in Unit 4: design a fault tolerance mechanism for the DFS.
Develop test-case scenarios to assess the system's resilience.

**Unit 5: Distributed Transaction and Distributed Shared Memory** (8+1 Hrs)

**PART A:** Distributed Transaction: Transaction Model, Classification, Implementation, Concurrency Control: Serializability, 2 Phase Locking, Strict 2 PL, Distributed Commit: 2 Phase Commit, Recovery


**PART B:** Distributed File Systems: Data Intensive Computing, Google FS, BigTable

**Text Books**

**Reference Books**

**Additional Reading**

**Course Outcomes:**

Upon completion of the course, graduates will be able to -

1. Develop the solutions for Communication, Synchronization, Mutual exclusion and Deadlock handling in distributed computing.

2. Identify the basic principles, design requirements and challenges associated with implementing large-scale distributed system.

3. Design of distributed application using modern tools in Distributed System.

4. Construct an optimal and cost-effective solution without compromising the security and reliability of the system with respect to convenience, efficiency and the ability to evolve.
5. Build a basic prototype distributed system for different applications in a team.

6. Apply principles of distributed computing while building appropriate variations of existing solutions to meet the development contexts.
CS40114:: BUSINESS INTELLIGENCE AND ANALYTICS

Credits: 03

Teaching Scheme: - Theory 3 Hrs/Week

Prerequisites: Database Management Systems.

Unit I: Introduction (8+1 Hrs)

Data Warehousing & Multi-Dimensional Modeling: What is a data warehouse, need for a data warehouse, four characteristics of a data warehouse, components/architecture of a data warehouse, data marts, building a data warehouse. OLTP vs OLAP.

Multidimensional Modeling: ER Modeling vs Multi-Dimensional Modeling, Data cubes, OLAP operations, lattice of cuboids, multi-dimensional schemas (star, snowflake, galaxy), facts, dimensions, concept hierarchies, types of dimensions, measures

Data Integration: different types of digital data and their sources, ETL (extract-transform-load)

Enterprise Reporting: Metrics, Measurement, Measures, KPIs, Dashboards, Reports, Scorecards

PART B: Examples of business analyses – funnel analysis, distribution channel analysis and performance analysis; levels of decision making (strategic, tactical and operational BI); types of facts; types of measures; different OLAP architectures

Unit II Descriptive Analytics - I (8+1 Hrs)

PART A: Data Preprocessing: mechanisms of data collection and challenges involved therein. Typical preprocessing operations: combining values into one, handling incomplete or incorrect data, handling missing values, recoding values, subsetting, sorting, transforming scale, determining percentiles, data manipulation, removing noise, removing inconsistencies, transformations, standardizing, normalizing - min-max normalization, z-score standardization, rules of standardizing data.

Data visualization: role of visualization in analytics, different techniques for visualizing data based on the nature of data and what kind of insights need to be drawn

Descriptive Statistics: role of statistics in analytics, types of data (scales of measurement - NOIR), data distributions, measures of variability (range, quartile, five number summary, variance, std dev, coeff of variation), analyzing distributions, Chebychev’s Inequality, measures of shape (skewness, kurtosis), measures of association (covariance, correlation), outliers

PART B: Data quality and its various aspects, measures of central tendency (arithmetic mean, geometric mean, harmonic mean, median, mode)
Unit 3: Descriptive Analytics - II (7+1 Hrs)
PART A : Inferential Statistics: Role of probability in analytics. Need for sampling, generating samples, sampling and non-sampling error. Sampling Distribution of Mean, Central Limit Theorem, Standard Error.

*Estimation*: Point and Interval Estimates, Confidence Intervals, level of confidence, sample size.

*Hypothesis Testing*: basic concepts, Errors in hypothesis testing, Power of test, Level of significance, p-value, general procedure for hypothesis testing. Parametric tests – z test, t test, chi-square test.  Hypothesis testing of means: two tailed and one-tailed tests. Chi-square test for independence and goodness of fit. Hypothesis testing for comparing two related samples. Limitations of hypothesis testing. Picking up the right test for a given scenario.

PART B : Common probability distributions and their characteristics, parametric vs non-parametric tests, ANOVA

Unit 4: Predictive Analytics (8+1 Hrs)
PART A : Similarity Measures: Design of recommender systems - user based and item based collaborative filtering

*Modelling*: Data Modelling Basics, Logic driven modeling, data driven modeling

*Regression Analysis*: Correlation and regression, Simple Linear Regression Model, Least Squares Method. Making Data Models more flexible, making data models more selective, dealing with Categorical variables, Interpretation of regression coefficients, fine tuning data models (assessing the fit, model fitting), Coefficient of determination, Significance tests, Residual analysis, Prediction intervals. Model evaluation techniques. Assumptions of regression analysis.

*Introduction to Time Series Analysis and Forecasting*: Time series patterns, forecast accuracy, moving averages and exponential smoothing, casual models, using regression analysis for forecasting, determining best forecast model to use, ARIMA models

PART B : Basic what-if spreadsheet models, different types of regression and which is used when, modeling non-linear relationships, heteroscedasticity, multicollinearity

Unit 5: Prescriptive Analytics & Emerging Trends in BI&A (8+1 Hrs)
PART A : Optimization Analytics: Overview of simulation and risk analysis, Linear Optimization Models (linear programming), Integer Linear Optimization models (integer programming), Non-linear optimization models (portfolio theory), Monte Carlo Simulation, Decision Analysis

*BigData Analytics*: What is Big Data, sources of BigData, MapReduce, Hadoop, statistical analysis of big data, visualizing bigdata

*Familiarity with the following areas*: multi variate analytics, text analytics, web analytics, social media analytics, in-memory analytics – row vs columnar databases, in-memory databases

PART B : NoSQL databases for unstructured data and its role in analytics, analytics in the cloud, embedded BI, mobile analytics, mobile BI, Data Stream Analytics, Customer Centricity, Patient Centricity, IoT, Self-Service BI, Fog networks

Text Books
1. “Business Analytics” by James R Evans, Pearson
Course Outcomes:

Upon completion of the course, graduates will be able to -

1. Construct an end-to-end data warehousing solution for business intelligence involving various data sources, ETL, multi-dimensional modeling, OLAP, reporting and analytics
2. Evaluate various data processing algorithms in their applicability to different problems
3. Display the process of converting data into a user defined format required for particular analysis
4. Utilize statistical tools in deriving insights from data
5. Describe various techniques for descriptive, predictive and prescriptive analytics
6. Apply various techniques to solve real-world data analysis problems
CS42105:: MOBILE COMPUTING

Credits: 03

Teaching Scheme: - Theory 3 Hrs/Week

Prerequisites:
- Communication Engineering
- Computer Network

Unit 1: Cellular Network (8+1 Hrs)
PART B: Signal and signal propagation, IEEE 802.11standard, 802.11a,b,g, 802.15.

Unit 2: GSM Communication (8+1 Hrs)
PART A: System Architecture: GSM Radio subsystem, Interfaces, Network and switching subsystem, Operation subsystem. GSM channels, GSM protocol architecture, Location tracking and call setup, Security, Data services N/W signaling, GSM mobility management, Administration and maintenance.
Handoff- Initialization of handoff, Delaying handoff, Forcing handoff, Power different handoff. Mobile assisted handoff, Intersystem handoff.
PART B: Survey of GSM network, Hard, Soft Handoff.

Unit 3: GSM Bearer Services (9+1 Hrs)
PART A: SMS architecture protocol, Hierarchy, VOIP services for mobile networks.
WAP: model and architecture, Gateway, protocol stack.
Telecommunication system: GPRS, wireless in local loop, DECT, EDGE, UMTS, Paging systems, CDPD.
PART B: Wireless application environment, Bluetooth.

Unit 4: Mobile Network and Transport layer (8+1 Hrs)
Mobile Transport layer: Traditional TCP, Indirect TCP, Snooping TCP, Mobile TCP, Fast retransmit/recovery, Transmission/time-out freezing, Selective retransmission, Transaction-oriented TCP.
PART B: TCP over 2.5/3G wireless networks.

Unit 5: Mobile Databases (7+1 Hrs)
PART A: Database hoarding, Data caching, Data cache and web cache maintenance in mobile environments, Client-Server computing and adaptation, Query processing, Data recovery process, Issues relating to quality of service, Digital audio broadcasting: DAB System, DAB objects, Object transfer protocol, DVB: DVB system.
PART B : Mobile Billing

Text Books

Reference Books

Course Outcomes:
Upon completion of the course, the students will be able to-
1. Describe the functional specification of 2G and 3G Cellular Network Standards.
2. Compute performance parameters for designing the Cellular Network.
3. Propose Telecommunication system to be deployed to fulfill bandwidth capacity planning.
4. Justify the Mobile Network performance parameters and design decisions.
5. Predict the requirements of next generation mobile network.
6. Design Mobile Application to solve a real world problem.
CS42131: Enterprise Systems

Credits: 03
Teaching Scheme: - Theory 3 Hrs/Week

Prerequisites: Software Engineering, Database Management System

Unit 1: Business Process Management (8 Hrs)

Process Orchestrations: Control Flow Patterns, Workflow Nets, Business Process Modeling Notation
Process Choreographies: Motivation and Terminology, Development Phases, Process Choreography Design and Implementation, Service Interaction Patterns
Properties of Business Processes: Data Dependencies, Structural Soundness, Soundness, Relaxed Soundness, Weak Soundness, Lazy Soundness, Soundness Criteria Overview

Part B: Petri Nets, Event-driven Process Chains, Yet Another Workflow Language, Graph-Based Workflow Language

Unit 2: SOA Fundamentals (8 Hrs)

Service Contracts (Standardization and Design): Contracts principles, Types of Service Contract Standardization, Contracts and Service Design, Versioning, Technology/Development Tool Dependencies

Part B: XML; HTTP; SOAP

Unit 3: SOA Design Principles (8 Hrs)

Part A: Service Abstraction (Information Hiding and Meta Abstraction Types): Abstraction principles, Types of Meta Abstraction, Measuring Service Abstraction, Service Abstraction and Service Design, Risks Associated with Service Abstraction
Service Reusability (Commercial and Agnostic Design): Reuse Principle, Service Reuse in SOA, Service Reusability and Service Design
Service Autonomy (Processing Boundaries and Control): Autonomy Principle, Types of Service Autonomy, Measuring Service Autonomy, Service Contract Autonomy (services with normalized contracts) Autonomy and Service Design
Service Statelessness (State Management Deferral and Stateless Design): State Management, Measuring Service Statelessness, Statelessness and Service Design
SOA Delivery Strategies, Service-Oriented Analysis: Introduction, Service Modeling, Service-Oriented Design: Introduction, SOA Composition Guidelines), Service Design

Part B: Importance of WSDL, SOAP, The use of registries via UDDI

Unit 4: SOA Technology and Implementation (8 Hrs)
Part A: Service Discoverability (Interpretability and Communication): Discoverability, Types of Discovery and Discoverability, Measuring Service Discoverability, Discoverability and Service
Service Composability (Composition Member Design and Complex Compositions): Composition, Composition Concepts and Terminology, Complex Service Composition, Measuring Service Composability and Composition Effectiveness Potential, Composition and Service Design, Service-Orientation and Object-Orientation, Mapping Service-Orientation Principles to Strategic Goals
Part B: SOA Platforms, SOA support in .NET and J2EE platforms

Unit 5: Enterprise Architecture (8 Hrs)

Text Books:

Reference Books:

Course Outcomes:

Upon completion of the course, graduates will be able to -

1. To model business requirements and business processes using BPMN 2.0 standard
2. To discover the set of services with composite services creation and designing services to facilitate integration and understand interrelationships among SOA, Web Services, OOD and an IT infrastructure
3. To focus Enterprise SOA and facilitate use of Enterprise Service Bus in an Enterprise SOA
4. To explore the concepts and technology for service orchestration and discuss the guidelines to integrate a Business Process Management Solution in an Enterprise SOA.
5. To practice the concepts, specifications, and technologies for service composition and explore the role of an enterprise service bus in an Enterprise SOA.
6. To understand case studies, lessons learned, and best practices for planning and implementing SOA projects
CS42127:: CLOUD COMPUTING

Credits: 03  
Teaching Scheme: - Theory 3 Hrs/Week

Prerequisites: Computer Networks, Operating Systems, Web Technologies

Unit 1 : Introduction  
(6+1 Hrs)


PART B : Study of top 5 cloud providers and top 5 applications on cloud

Unit 2 : Architecture and IaaS  
(8+1 Hrs)

PART A : Cloud Computing Architecture: Cloud computing stack - Comparison with traditional computing architecture (client/server), Services provided at various levels, How Cloud Computing Works, Role of Networks in Cloud computing, protocols used, Role of Web services. Service Models (XaaS) - Infrastructure as a Service(IaaS), Platform as a Service(PaaS), Software as a Service(SaaS). Deployment Models, Public cloud, Private cloud, Hybrid cloud, Community cloud


PART B : Study top 5 open source tools for building private cloud along with pros/cons

Unit 3 : PaaS and SaaS  
(7+1 Hrs)

PART A : Platform as a Service (PaaS): Introduction to PaaS - What is PaaS, Service Oriented Architecture (SOA). Cloud Platform and Management - computation, storage

Software as a Service (SaaS): Introduction to SaaS, Web services, Web 2.0, Web OS, Case Study on SaaS

PART B : Innovative applications of cloud computing

Unit 4 : Cloud Management and Security  
(10+1 Hrs)
PART A: Service Management in Cloud Computing: Service Level Agreements (SLAs), Billing & Accounting, Comparing Scaling Hardware: Traditional vs. Cloud, Economics of scaling: Benefitting enormously, Managing Data - Looking at Data, Scalability & Cloud Services, Database & Data Stores in Cloud, Large Scale Data Processing


PART B: Study the cloud deployment of a big enterprise

Unit 5: Cloud Solutions (9+1 Hrs)

PART A: Case study on Open Source and Commercial Clouds – Amazon EC2, Google Compute Engine, Microsoft Azure, Cloudfoundry, OpenStack

PART B: How the CDNs (Content Delivery Networks) make use of the cloud?

Text Books
3. Cloud Security by Ronald Krutz and Russell Dean Vines, Wiley-India

Reference Books
2. Cloud Security & Privacy by Tim Malhar, S. Kumaraswammy, S. Latif (SPD, O’REILLY)
4. Cloud Computing Bible by Barrie Sosinsky, Wiley India.
5. Cloud Computing, Michael Miller, Que Publishing

Course Outcomes:
Upon completion of the course, graduates will be able to:

1. Describe the main concepts, key technologies, strengths, and limitations of cloud computing and the possible applications for state-of-the-art cloud computing
2. Explain the architecture and infrastructure of cloud computing, including SaaS, PaaS, IaaS, public cloud, private cloud, hybrid cloud, etc.
3. Collaboratively research and write a paper on the state of the art (and open problems) in cloud computing.
4. Identify problems, and explain, analyze, and evaluate various cloud computing solutions.
5. Choose the appropriate technologies, algorithms, and approaches for the related issues.
6. Display new ideas and innovations in cloud computing.
CS42103:: PARALLEL COMPUTING ON GPU

Credits: 03

Teaching Scheme: - Theory 3 Hrs/Week

Prerequisites:

Unit 1: Parallel programming basics (8+1 Hrs)
PART A: Parallel programming definition, motivation, Types and levels of parallelism, Different grains of parallelism, data dependence graph, data parallelism, functional parallelism, Flynn’s classification of multi-processors, Motivation for heterogeneous programming, Definition of thread and process, programming parallel computers- extend a compiler, extend a sequential programming language, add a parallel programming layer, create a parallel language.
PART B: multiprocessor and multicomputer systems, interconnection networks

Unit 2: Introduction to GPU, architecture and parallel algorithms (8+1 Hrs)
PART A: Introduction to GPU computing, motivation, Modern GPU architecture case study: NVIDIA Fermi Tesla C2050/Kepler K20; GPU memories- global, shared, texture memory and their properties and uses, roles of CPU and GPU in parallel computing, GPU computing domain areas and success. Parallel algorithm design. Speedup and scalability.
PART B: parallel algorithm design for data clustering, theory of locality of reference

Unit 3: Compute Unified Device Architecture (CUDA) (8+1 Hrs)
PART A: CUDA Architecture, CUDA programming model, execution model, thread organization: Concept of grid, block and thread, thread index generation, warp; memory model: Introduction to global, shared, local memories, usage of cache, texture cache, constant memory, memory banks and bank conflicts, memory coalescing,, CUDA structure and API details. CUDA example programs (Vector dot product, Vector-Matrix multiplication and etc).
PART B: atomic operations in CUDA

Unit 4: Problem solving using GPUs (8+1 Hrs)
PART A: Single vss double precision, solving problems that involves Vectors, Matrices, Binomial coefficients, Bernstein coefficients and etc. Instructor will choose the problems from several domains.
PART B: Study problems given by instructor

Unit 5: Parallel reduction and Tools (8+1 Hrs)
PART A: Reduction operation using prefix sum example. Performance issues in algorithms- deciding parallelization of a part of algorithm and selecting the highest parallelism, Need of profilers, Introduction to CUDA Tools: MemCheck and & Visual Profiler.
PART B: Memory leaks and associated problems
Text Books
2. CUDA by Example: An Introduction to General-Purpose GPU Programming by Jason Sanders and Edward Kandrot
3. Parallel Programming in C with MPI and OpenMP by Michael J. Quinn, Tata McGraw-Hill Edition

References:
2. www tutorials on introduction to parallel computing
3. Other references suggested by instructor

Course Outcomes
Upon completion of the course, graduates will be able to -

1. Analyze the real problem for exploiting maximum parallelism on GPU architecture
2. Solve the complex problems using GPUs
3. Compare serial and parallel executions.
4. Code and optimize the parallel programs on GPU using CUDA.
5. Apply parallel computing methods to research oriented problems.
6. Evaluate success of CUDA projects
CS42132: NETWORKS SECURITY

Credits: 03  
Teaching Scheme: - Theory 3 Hrs/Week

Prerequisites: Computer Networks.

Unit 1: Introduction (8+2 Hrs)
PART B. SYN-Flooding, SQL-injection, DNS poisoning, Sniffing, Sabotage flashing attack, Security of Embedded Devices.

Unit 2: Private key cryptography (8+2 Hrs)
PART A : Mathematical background for cryptography: modulo arithmetic, GCD (Euclids algorithm), algebraic structures (Groups, Rings, Fields, Polynomial Field). Role of random numbers in security, Importance of prime numbers Data Encryption Standard: Block cipher, Stream cipher, Feistel structure, round function, block cipher modes of operation, S-DES, Attacks on DES, S-AES, AES.
PART B : Chinese remainder theorem, Elementary Ciphers (Substitution, Transposition and their Properties), Frequency analysis

Unit 3: Public key cryptography (8+2 Hrs)

Unit 4: Authentication and access control (8+2 Hrs)
PART B : Authentication and authorization tools, Biometrics.
Unit 5: Security application and design  

(8+2 Hrs)

PART A : Network layer security: IPSec for IPV4 and IPV6.
Transport layer security: SSL and TLS.
Application layer security: Security services, S/MIME, PGP, PEM, Https, Honey pots.
Security design: End-to-end security, Security composability, Open design, Cost and tradeoffs

PART B : Cyber laws, Cyber Security, IDS, SNORT, Firewall, Cloud Security
Tradeoffs.

Text Books

Reference Books

Additional Reading

Course Outcomes

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

1. Analyze cryptographic techniques using a mathematical approach by examining nature of attack.
2. Establish type of attack on a given system.
3. Simulate different types of attacks using tools.
4. Justify various methods of authentication and access control for application of technologies to various sections of industry and society.
5. Design a secure system for protection from the various attacks for 7 layer model by determining the need of security from various departments of an organization.
6. Estimate future needs of security for a system by researching current environment on a continuous basis for the benefit of society.
CS42101:: ADVANCED COMPUTER GRAPHICS

Credits: 03  Teaching Scheme: - Theory 3 Hrs/Week

Prerequisites: Computer Graphics

Unit 1 : OpenGL
PART A. OpenGL over windows, SDK, Extensions, GLUT, GLU, OpenGL primitives, Programming language: Blending, 3D viewing(camera analogy), Lighting model, Culling, Fog, Texture mapping.
PART B. OpenGL over Linux, pBuffer rendering, Shadowing Techniques.

Unit 2 : Graphics Hardware
PART A. Graphics card, Components of graphics card, General-Purpose computation, GPU, GPGPU, CUDA architecture, CUDA Programming: Type qualifiers, Execution configuration, vector types, functions, APIs.
PART B. How graphics card works? GeForce 6800 series GPU architecture.

Unit 3: Advanced Rendering Techniques
PART A. Point based rendering, Mesh Simplification, Spatial partitioning, Solid Modeling, Subdivision surfaces; Catmull-Clark subdivision, Subdivision rules, Visibility Computation; culling types, cells and portals, hardware support.
PART B. Splines , Tessellation, 3D viewing.

Unit 4: Photorealistic and Volume Rendering
PART B. Monte Carlo mathematical formulation, Marching cubes algorithm.

Unit 5: Texture Synthesis and Image Processing
PART A. Texture synthesis, Image processing: Digital image representation, Image data structures, Sampling and Quantization, Image enhancement in spatial domain.
PART B. Image compression, Image synthesis.

Text Books

Reference Books

Additional Reading

Course Outcomes:
Upon completion of the course, graduates will be able to -

1. Design algorithms/programs for 2D/3D graphics using OpenGL.
2. Discriminate data structures and algorithms of 3D modeling, visibility and rendering.
3. Describe graphics hardware for its functions and programming.
4. Distinguish algorithms of global illumination and 3D volume rendering.
5. Conform the use of image and texture processing for graphics.
6. Construct creative 2D/ 3D graphics models.
CS42125:: Randomized and Approximation Algorithms

Credits: 03  
Teaching Scheme: - Theory 3 Hrs/Week

Unit 1: Basic probability theory  
Part A: Introduction to randomization in computation and some simple randomized algorithms. **Basic discrete probability theory:** basic counting, definition of probability, examples, independence of events, conditional probability, union bound, inclusion exclusion, Bayes’ rule, discrete random variables, expectation, variance, linearity of expectation, sum of independent random variables, standard distributions (Bernoulli, Binomial, Geometric), coupon collector problem, birthday paradox, probabilistic recurrences.  
Uniform generation of combinatorial structures. Indicator random variables and their role in algorithm analysis.  
Part B: Review Las Vegas and Monte-Carlo algorithms (with examples: randomized quick sort, Karger’s min-cut algorithm.)

Unit 2: Tail inequalities, randomized complexity classes  
Part A: Tail Inequalities and applications  
Moments and deviation, occupancy problem, Markov and Chebyshev inequalities and some applications, randomized selection, weak law of large numbers, stable marriage problem and principle of deferred decision, coupon collector problem and sharp threshold, Chernoff’s bound and some applications, set balancing.  
Complexity classes  
Basic complexity classes P, NP, RP, Co-RP, ZPP, BPP and their interrelations, probability amplification in RP and BPP, randomness and nonuniformity, Adleman’s theorem.  
Part B: Yao’s min-max principle and lower bound for randomized computations.

Unit 3: Algebraic techniques  
Part A: Polynomial identity testing, Schwartz-Zippel lemma and applications (with examples verifying matrix multiplication, testing equality of strings, perfect matching problem for bipartite graphs), Mulmuley-Vazirani-Vazirani isolation lemma and application to matching problem. Number theoretic algorithms (finding quadratic non-residues, primality testing), introduction to probabilistic methods.  
Part B: Application of randomized algorithms in geometric problems (Convex hulls, half space intersection, Delaunay triangulations, diameter of point set).

Unit 4: Markov Chains and Random Walks  
Part A: Markov chains: definition, representations, randomized algorithm for 2-SAT and 3-SAT, classifying states of Markov chains, Gambler’s ruin, stationary distributions. Random walks on undirected graphs, cover time, hitting time, commute time, graph connectivity, electrical networks, introduction to expander graphs.  
Part B: Expanders and rapidly mixing random walks.
Unit 5: Approximation Algorithms  

Part A: Introduction to approximation algorithms, NP-hard optimization problems, lower bounding OPT, example of set-cover (O(\log n) factor approx-algorithm based on greedy strategy, layering), Shortest super-string problem, Knapsack and FPTAS algorithms.

Linear programming based algorithms, LP relaxation, LP duality. LP rounding strategy and primal-dual schema, set-cover and some other examples using LP based techniques, maximum satisfiability.

Part B: Review Vertex cover problem, Traveling Salesman Problem (TSP), general TSP hard to approximate, metric TSP.

Advanced Topics: 
(depending on the availability of time some of the following topics may be included)
Role of expander graphs in derandomization, hardness of approximation, statement of PCP theorem, approximation algorithms for classical problems on integer lattices.

Text books:

2. Probability and computing by Michael Mitzenmacher, Eli Upfal
3. Approximation Algorithms by Vijay V. Vazirani (Springer)

References:

3. For some of the topics some online lecture-notes or original papers are used as reference.

Course Outcomes:

Upon completion of the course, graduates will be able to -

1. To solve problems based on the basic discrete probability and combinatorics
2. To design Las-Vegas, Monte-Carlo randomized algorithms for various computational problems
3. To analyze time complexity and success probability of randomized algorithms using random variables.
4. To illustrate application of tail inequalities in tight estimation of the success probability and the time complexity of randomized algorithms
5. To explain role of advanced algebraic techniques such as Schwartz-Zippel Lemma, Isolation Lemma, Markov chains and random walks on graphs in randomized algorithms design
6. To design approximation algorithms for NP-complete problems using suitable paradigm
CS42113:: DIGITAL SIGNAL PROCESSING

Credits: 03

Teaching Scheme: - Theory 3 Hrs/Week

Prerequisites:

Unit 1: Introduction to signals and systems (8+1 Hrs)
PART B. Properties of LTI systems, parallel and cascade connection, Correlation of

Unit 2: Z and Fourier transforms (8+1 Hrs)
PART B: Symmetry properties of F. T, F. T. theorems: Linearity, time shifting, frequency shifting, time reversal, differentiation.

Unit 3: Frequency analysis of Signals and Systems (8+1 Hrs)
PART B : Properties of DFT: Linearity, DIF FFT, Goertzel Algorithm, Inverse DFT using FFT(DIT)

Unit 4: Design of Digital Filters (9+1 Hrs)
PART A: Concept of filtering, Ideal filters and approximations, specifications, IIR filter design from continuous time filters: Characteristics of Butterworth, Chebybyhev approximations, impulse invariant and bilinear transformation techniques, Design examples, FIR filter design using windows: properties of commonly used windows,
systems with Linear phase, Generalized Linear phase systems, Four Types of GLPS (Type I), Design Examples, Design using Kaiser window, Comparison of IIR and FIR Filters.

B. Four Types of GLPS (Type II, III, IV), Examples on Filter Design (IIR & FIR)

Unit 5: Realization of Filters (7+1 Hrs)


PART B: Instruction set of ADSP 21XX series processor and some examples.

Text Books

Reference Books

Course Outcomes:
Upon completion of the course, graduates will be able to -
1. Convert analog signal into a digital signal without irreversible data loss using mathematical techniques.
2. Design faster algorithms for signal domain conversion which ensures expected operation on all architectures.
3. Design filters meeting the given specifications with the help of Matlab.
4. Design a stable system which requires minimum components to implement.
5. Analyse a signal in different mathematic domains to understand the signal characteristics.
6. Differentiate between various realizations techniques.
CS42119:: INFORMATION RETRIEVAL

Credits: 03

Teaching Scheme: - Theory 3 Hrs/Week

Prerequisites: Data Structures, Database Management Systems

Unit 1: Information Retrieval Introduction and Models (9 Hrs)

PART A: Introduction, Definition, Objectives, Search and Browse Capabilities; A Formal Document Representation, Characterization of IR Models, Boolean Retrieval, Extended Boolean Models, Vector Space Model, Probabilistic Model, Measure of Relatedness, Term Weighting, Naive Bayes Text Classification, Document and Term Clustering, Flat and Hierarchical Clustering, Matrix Decomposition, Latent Semantic Indexing

PART B: Bayesian Model, Data Structures and Techniques for Ranking, Models for Browsing, Zipf's law, Heap's law

Unit 2: Query Processing and Retrieval Evaluation (8 Hrs)

PART A: Digital libraries, Morphological, Lexical Analysis, Stemming Algorithms, Thesaurus Construction, Ontology, Metadata, Query Languages, Similarity Measures and Ranking, Relevance Feedback, Query Expansion, Retrieval Performance, Evaluation Measures for Ranked and Unranked Results

PART B: Porter's Stemming Algorithm, Automatic Local/Global Analysis, Information Summarization and Visualization, Archiving and Preservation

Unit 3: Indexing and Searching (8 Hrs)

PART A: Automatic Indexing, Inverted Files, Structures Used, Signature Files, Compression, Partitioning, Tries, Suffix Trees and Suffix Arrays, Index Construction, Distributed Indexing, Index Compression, Sequential Searching, Pattern Matching, String Matching allowing Errors, Regular Expressions and Extended Patterns, Pattern Matching using Indices, Structural Queries

PART B: Fast Inversion (FAST-INV) Algorithm, Algorithms on PAT Tree, Faceted Search

Unit 4: Parallel, Distributed IR and Web Searching (8 Hrs)

PART A: Parallel IR, Index Construction, Distributed IR, Characterizing the Web, Search Engines, Browsing, Metasearchers, Searching using Hyperlinks, Crawling, Link Analysis, Architectures (Agents, Buses, Wrappers/Mediators)

PART B: Watermarking, PageRank Algorithm, HillTop Algorithm

Unit 5: Multimedia IR (7 Hrs)

PART A: Multimedia Data Modeling, Query Languages, A Generic Multimedia Indexing (GEMINI) Approach, One Dimensional Time Series, Two Dimensional Color Images, Automatic Feature Extraction, Operations on images, Motion detection, Object recognition, Automatic image annotation and retrieval, Audio, Graph, Video Retrieval
PART B: Hashing Algorithms, Image Features and Similarity Functions

Text Books

Reference Books
3. “Information Retrieval”, C. J. Van Rijsbergen, Information Retrieval Group, University of Glasgow, online at http://www.dcs.gla.ac.uk/Keith/Preface.html

Course Outcomes:
Upon completion of the course, graduates will be able to -
1. Construct various indexes using suitable techniques
2. Validate retrieval performance of an information retrieval system
3. Apply sequential search and pattern matching techniques
4. Describe various models for information retrieval system
5. Illustrate working of parallel, distributed and multimedia information retrieval system
6. Use various information retrieval algorithms and different types of queries to address real world challenges
CS42114: PRODUCT DESIGN

Credits: 03  
Teaching Scheme: - Theory 3 Hrs/Week

Prerequisites:

Unit 1: Introduction and User Studies  (9+1 Hrs)
PART B: Selection of a Product for Design Study such as Antivirus, Operating System, Mobile Phones, Web Browsers, Accounting Software etc.

Unit 2: Design and Usability Evaluation  (7+2 Hrs)
PART B: Ethnographic and Cultural Study of Selected Product.

Unit 3: Categorization of Products  (8+2 Hrs)
PART A: Products for Future Use, Products to be Used in Groups, Devices used in Public Places, Products that Enrich User Experience, Embedded Products, Designer Products, Interfaces, Complexity of Interfaces, Design of Multi-Modal Interfaces, Expressive Interfaces, Natural Interfaces, Tangible Interfaces, Faulty Interfaces.
PART B: Classification of Selected Products and Possible Variances with Extended Features.

Unit 4: Design Management and Professional Practice  (8+2 Hrs)
PART B: Identification of IPR (Copyrights, Patents and Trademarks) Issues with Selected Products.
Unit 5: Product Life and Marketing (8+1 Hrs)

PART B: Analysis of Prices and Related Pricing Policy for Selected Product.

Text Books

Reference Books

Additional Reading

Course Outcomes:

Upon completion of the course, graduate will be able to -

1. Develop an IT product concept by applying suitable design principles, models and guidelines along with creativity.
2. Propose an implementation-centric product design with due consideration to ethnographic, cultural and accessibility aspects
3. Formulate all possible vital product features required for a specific category of product
4. Propose effective pricing policy along with multilingual documentation and support for the product
5. Build a set of skills required for responsible positions such as Product Designer, Interaction Designer and Creative Technologist
6. Follow required processes and standards while designing IT products
CS42115:: HUMAN COMPUTER INTERACTION

Credits: 03

Teaching Scheme: - Theory 3 Hrs/Week

Prerequisites:

Unit 1 : Introduction (7+1 Hrs)


PART B: Identification of Application Category and Related Features for Selected Product / System.

Unit 2 : Principles, Models & Guidelines (9+2 Hrs)


PART B: Task / Error Analysis for Selected Product / System.

Unit 3: Design Process and Interaction Styles (8+2 Hrs)


PART B: UI Design for Selected Product/System.

Unit 4: Evaluation Techniques and Interface Categories (8+2 Hrs)


PART B: Usability Evaluation of Selected Product/System.

Unit 5: Documentation and Groupware (8+1 Hrs)


B. Documentation Design for Selected Product/System.
Text Books

Additional Reading

Course Outcomes:

Upon completion of the course, graduates will be able to -

1. Identify human factors and usability issues related with computing applications
2. Differentiate computing applications into categories based on human factors
3. Design a user interface by applying suitable design principles, models and usability guidelines
4. Integrate ethno-cultural and accessibility computing aspects into the user interface design.
5. Display the impact of usability evaluation and testing in computing applications
6. Follow required processes and standards while designing user interfaces
CS42134: Modeling and Simulation

Credits: 3  
Teaching Scheme: 3 Hours / Week

Unit 1: Process of Modeling and Simulation  
Part A: What is M&S, Need for Abstraction, Relationship between modeling and simulation Process of modeling: Problem identification and formulation, Real system data collection, Model development, Validation, Experiment design, Simulation runs and Results interpretation.
Part B: Application areas: optimization, decision making support, forecasting, safety considerations, training and education

Unit 2: Formal models and modeling techniques  
Part A: Monte Carlo methods, Stochastic processes, Queuing theory: Little's Theorem and applications, M/M/1 Queuing System, Markov Chain Formulation, the M/M/m, M/M/m/m and other Markov Systems, Non-markovian queue, Analytical Model Vs. Simulation model, Petri nets and colored Petri nets, Game theory, State spaces and transitions.
Part B: Graph structures: directed graphs, trees, networks

Unit 3: Discrete Event Simulation  
Part A: Deterministic vs. stochastic simulation, Static vs. Dynamic Simulation, Constructing dynamic stochastic simulation models, Time keeping, Event Scheduling, State transition, Time driven and event driven models, Pseudo-random number generation.
Part B: SimPy basics

Unit 4: Agent-based simulation  
Part B: NetLogo basics

Unit 5: Advanced Topics  
Part A: Intro to Parallel Discrete Event Simulation: Parallel and Distributed Platforms, Model scalability, Virtual Reality, Virtual Worlds, Intro to Rare Event Simulation.
Part B: PDES Challenges

Text Books:

Reference Books:


**Course Outcomes:**
Upon completion of the course, graduates will be able to –
1. Develop a model for a given problem using appropriate modeling and simulation technique/formalism.
2. Implement discrete event simulation models using general-purpose programming languages or DES frameworks.
3. Demonstrate the effectiveness of modeling and simulation at predicting behavior/performance/problems of systems under development.
4. Design an agent-based simulation model for a complex system.
5. Contribute towards increased utilization of modeling and simulation as a problem solving approach for issues in governance and industry where it could be applied.
6. Adapt to the changing needs of the organizations and individuals during the development process.
CS40215:: DISTRIBUTED COMPUTING

Credits: 01

Teaching Scheme: -Tutorial 1 Hr/Week

Prerequisites: Operating Systems.

List of Contents

A TERM-WORK containing the record of the following:

1. Design and develop a basic prototype distributed system (e.g. a DFS).
2. Design and implement client server application using RPC/ RMI mechanism (Java)
3. Design and implement a clock synchronization algorithm for prototype DS.
4. Implement Ring or Bully election algorithm for prototype DS.
5. Ricart Agrawala’s distributed algorithm for mutual exclusion.
7. Simulate Wait for Graph based Centralized or Hierarchical or Distributed algorithm for deadlock detection.
8. Implementation of 2PC / Byzantine Generals Problem
9. Simulate any one of the Consistency models.

Text Books

Reference Books
Additional Reading
CS42125:: Randomized and Approximation Algorithms

Credits: 01

Teaching Scheme: - Tutorial 1 Hr/Week

Prerequisites: Prior introduction to basic probability theory is helpful though it is not presumed. First course on algorithms.

List of Contents

A TERM-WORK containing the record of the following:

Assignments:

1. Problem solving based on basic probability (random variables, mean, variance)
2. Problem solving based on basic probability (linearity of expectation, some applications, conditional probability)
3. Use of indicator random variables in algorithm analysis
4. Tail Inequalities and applications (Markov, Chebyshev)
5. Tail Inequalities and applications (Chernoff bound, variance of sum of co-related random variables)
6. Random walks, rapid mixing and some applications like sampling points from convex bodies
7. Schwartz-Zippel Lemma and applications
8. Complexity of perfect matching problem
9. Randomization in Number theoretic and geometric algorithms
10. Approximation algorithms, lower bounding OPT
11. Linear programming and LP-duality, LP based approximation algorithms
12. PCP theorem and hardness of approximation
Text Books
2. Approximation Algorithms by Vijay V. Vazirani (Springer)

Reference Books
CS42113:: DIGITAL SIGNAL PROCESSING

Credits: 01

Teaching Scheme: - Tutorial 1 Hrs/Week

Prerequisites:

List of Tutorials

1. Sampling theorem: Write a program in Matlab to verify sampling theorem

2. Linear Convolution: Write a program in Matlab to perform linear convolution of 2 user defined signals. Verify your answer using Matlab functions.

3. Cross co-relation: Get two signals from user and write a code in Matlab to perform cross co-relation. Verify your answer using Matlab functions.

4. Auto co-relation: Get a signal from user and write a program in Matlab to verify the following two properties of the Auto co-relation.
   A. Auto co-relation attains its maximum value at zero lag
   B. Auto co-relation is an even function

5. Pole Zero: For a given Z transform of signal, plot poles and zeroes on Z space using Matlab and verify that when for a system, if pole is near unit circle, rate of decay of the response in time domain is much more gradual than a system with pole near origin.

6. Magnitude and Phase response of a system: For a given transfer function, plot magnitude and phase response of the system using Matlab

7. DFT: Write a program to calculate 8 point DFT of a user defined signal using Matlab. Verify your answer using Matlab functions.

8. Circular Convolution: Get two signals from user and write a program in Matlab to calculate their circular convolution. Verify your answer using Matlab functions.

9. FFT: Write a program to implement a Radix 2, 8 Point DIT FFT algorithm using Matlab and verify your answer using in built functions.

10. IIR Filter design: Get the filter specification from the user and write a program to calculate the filter coefficients in C/C++
11. FIR Filter design: Get the filter specification from the user and write a program to calculate the filter coefficients in C/C++

Text Books

Reference Books
CS42219:: INFORMATION RETRIEVAL

Credits: 01  
Teaching Scheme: - Tutorial 1 Hrs/Week

Prerequisites: Data Structures, Database Management Systems

List of Contents

A TERM-WORK containing the record of the following:

A. Assignments :

1. Problem solving for Boolean model.

2. Problem solving for Vector Space model.

3. Study of Stemming Algorithms and Thesaurus Construction

4. Problem solving for Index creation: Inverted Files, Signature Files, Suffix Trees and Suffix Arrays

5. Problem solving for Sequential Searching and Pattern Matching techniques

6. Problem solving for Latent Semantic Indexing for Text Classification

7. Build an Information Retrieval system using Apache Lucene

Text Books


Reference Books


3. Apache Lucene Tutorial
CS42114:: PRODUCT DESIGN

Credits: 01

Teaching Scheme: - Tutorial 1 Hr/Week

Prerequisites:

List of Contents
A TERM-WORK containing the record of the following:
1. Design a questionnaire for study of selected product.
2. Study specialized user population and their impact on the design of selected product.
3. Perform a GOMS analysis for any task(s) related with selected product.
4. Study faulty interfaces/interactions related with selected product.
5. Analyze the cybercrimes related with selected product and suggest preventive measures.

Text Books

Reference Books

Additional Reading
CS42115:: HUMAN COMPUTER INTERACTION

Credits: 01  
Teaching Scheme: - Tutorial 1 Hr/Week

Prerequisites: :

List of Contents

A TERM-WORK containing the record of the following:

1. Identify specialized users and related facilities for a selected product / system and make necessary suggestions for its improved accessibility design.

2. Design user persona for the users of selected product / system.

3. Conduct a contextual inquiry for selected product/system.

4. Design an interface prototype for selected product/system.

5. Evaluate an interface using usability evaluation technique.

Text Books


CS42234: Modeling and Simulation

Credits: 1                                          Teaching Scheme: Tutorial 1 Hours / Week

List of Contents:

1. Develop a simple deterministic simulation to determine the loan tenure for Rs. X principal amount when the customer pays Rs. Y per month. Assume the fixed interest rate of 10% per year.

2. Develop a Monte Carlo simulation model for profit estimation before introducing a new product in the market. Consider the uncertainty in terms of sales, reduction costs, competitive pricing and other market dynamics.

3. Develop a discrete event simulation of a typical fast-food restaurant. Restaurant configuration, business factors and customer behaviour factors should be tunable parameters.


5. Develop a parallel discrete event simulation for a network of routers using conservative event processing.

Text Books:


Reference Books:

CS40314:: BUSINESS INTELLIGENCE AND ANALYTICS

Credits: 01  Teaching Scheme: -Practical 2Hr/Week

Prerequisites: Database Management Systems.

List of Contents

A TERM-WORK containing the record of the following:

1. Getting started: understand what business do with their data with a scenario based application using QlikView tool
2. Designing an end to end warehousing solution for a real world scenario involving multi-dimensional modeling, designing data cube, doing ETL, OLAP and reporting
3. Getting started with R
4. Using R for data preprocessing, exploratory analysis, visualization, correlation and regression analysis, hypothesis testing, chi square test
5. Data analysis case study using R for a readily available data set
6. BigData Analytics - MapReduce and exposure to Hadoop, Using R over Hadoop
7. [Optional] A group mini-project: take a real world data analysis problem and solve it using the above learned concepts
   a. Getting Data from varied sources
   b. Data massaging to prepare it for analysis
   c. Generating visualizations to interpret descriptive analysis
   d. Implementing sampling and estimation techniques
   e. Regression analysis on data
   f. Hypothesis testing

Text Books
1. “Data mining and business analytics with R”, Johannes Ledolter, Wiley
2. “Business Analytics” by James R Evans, Pearson

Reference Books

1. “Business Intelligence for Dummies”
5. “Information Dashboard Design: The effective visual communication of data”, Stephen Few, O’Reilly
CS42305:: MOBILE COMPUTING

Credits: 01  Teaching Scheme: - Laboratory 2 Hrs/Week

Prerequisites: Computer Networks

List of Practical

Assignments should be implemented on android operating systems.

1. Design simple GUI application with activity and intents e.g. calculator.
2. Design an application to handle address book and call log.
3. Design an application to display map with current location.
4. Design an application to send SMS and emails.
5. Design a client server application to communicate with database server.

Course Project: The project needs to be done in the group of 2 or 3 where students need to select real world problem and present a solution. During the evaluation of the project the emphasis is on: (a) how problem selection is done and (b) is the application user friendly.

Text Books

Reference Books
CS42331::Enterprise Systems

Credits: 01  
Teaching Scheme: - Laboratory 2 Hrs/Week

Prerequisites: Software Engineering, Database Management System

List of Practical

2. To decompose and organize the problem domain area into broad subject areas and identify the boundaries of problem/system along with identification of Business Processes and develop full detail Business Process diagrams.
3. To develop Domain-driven vocabulary of the target system indicating domain lexicon and context-based terminologies.
4. To identify and categorize the target system services with detailed service specifications modeled with component diagram incorporating appropriate architectural style and coupling.
5. To design the service layers and tiers modeled with deployment diagram accommodating abstraction, autonomy, statelessness and reuse.
6. To map the service levels and primitives to appropriate Strategies for data processing using XML / XQuery/ JSON / JAXB.
7. To produce, invoke, compose Web Services using SOAP, WSDL and UDDI.
8. To implement and integrate the components of the target system using .NET / J2EE platforms adhering to Service specifications.
9. To create the balanced scorecard for the target system indicating the standards and principles applied.

Text Books:


Reference Books:

CS42327 :: CLOUD COMPUTING

Credits: 01  
Teaching Scheme: - Laboratory 2 Hrs/Week

Prerequisites: Computer Networks, web Technology, Operating Systems.

List of Contents

A TERM-WORK containing the record of the following:

1. Hands on virtualization using XenServer
2. Hands on containerisation using Docker
3. Deployment and Configuration options in Amazon (AWS)
4. Deployment and Configuration options in Google Cloud
5. Deployment and Configuration options in Microsoft Azure
6. Building a 'HelloWorld' app for the cloud
7. Deploying the 'HelloWorld' app for the cloud

Text Books
   2. Enterprise Cloud Computing by Gautam Shroff, Cambridge
   3. Cloud Security by Ronald Krutz and Russell Dean Vines, Wiley-India

Reference Books
   1. Google Apps by Scott Granneman, Pearson
   2. Cloud Security & Privacy by Tim Malhar, S.Kumaraswammy, S.Latif (SPD,O’REILLY)
   4. Cloud Computing Bible by Barrie Sosinsky, Wiley India
CS42303:: PARALLEL COMPUTING ON GPU

Credits: 01

Teaching Scheme: - Laboratory  2 Hrs/Week

Prerequisites:

List of Practical

1. Parallel GPU implementation of vector-vector operations
2. Parallel GPU implementation of vector-Matrix operations
3. Parallel computation of binomial coefficient matrix
4. Parallel GPU implementation of Matrix-Matrix operations
5. Assignment focusing on optimization of data transfer between CPU and GPU: using page locked host memory and to avoid the data transfer
6. Assignment focusing on memory optimization: use of GPU shared, constant and texture memory.
7. Parallel GPU implementation involving kernel looping.
8. Parallel computation of set of multi-indices on GPU.

Text Books

2. CUDA by Example: An Introduction to General-Purpose GPU Programming by Jason Sanders and Edward Kandrot
3. Parallel Programming in C with MPI and OpenMP by Michael J. Quinn, Tata McGraw-Hill Edition

References:

http://developer.nvidia.com/
www tutorials on introduction to parallel computing
Other references suggested by instructor
CS42312: NETWORKS SECURITY

Credits: 01

Teaching Scheme: - Laboratory 2 Hrs/Week

Prerequisites:

List of Practicals

1. Demonstrate: SQL injection, Cross-site scripting, buffer overflow.
2. Implement packet sniffer.
3. Implementation of Caesar and Vigenere Cipher
4. Implementation of Playfair Cipher
5. Implementation of Hill Cipher
6. Implementation of S-DES.
7. Implementation of S-AES
8. Implementation of RSA.
9. Implementation of Diffie-Hellman key exchange
10. Implementation of ECC algorithm.

Text Books


Reference Books


Additional Reading

CS42301:: ADVANCED COMPUTER GRAPHICS

Credits: 01

Teaching Scheme: - Laboratory 2 Hrs/Week

Prerequisites:

List of Practical

1. Implement an OpenGL program to draw different 2D shapes.
2. Implement an OpenGL program to draw 2 overlapped shapes and use alpha blending.
3. Implement an OpenGL program to draw 3D cube and apply transformations.
4. Implement an OpenGL program to draw 12 spheres and apply different light effects.
5. Implement an OpenGL program to draw scene and apply fog effect.
6. Implement an OpenGL program to draw 3D cube and apply different textures on different faces.
7. Implement CUDA program for the prefix addition.
8. Implement CUDA program for the multiply two matrices.
10. Implement a program for edge detection using Gaussian filter.

Text Books

Reference Books

Additional Reading
CS47303:: PROJECT STAGE 2

Credits: 04
Teaching Scheme: - Practical 8 Hrs/Week

Aim
This course addresses the issues associated with the successful management of a software development project. The course emphasizes project life cycle phases requirement engineering, system analysis and system design. A further aim is for students to heighten personal awareness of the importance of developing strategies for themselves and their career. The Project Work can lead to:

a. Transform existing systems into conceptual models.
b. Transform conceptual models into determinable models.
c. Use determinable models to obtain system specifications.
d. Select optimum specifications and create physical models.
e. Apply the results from physical models to create real target systems.

Overview of the Course:

1. The Student Project Group is expected to make a survey of situation for identifying the requirements of selected Technological Problem. The Student Project Group will be monitored by Internal Guides and External Guides (if any).
2. The project requires the students to conceive, design, implement and operate a mechanism (the design problem). The mechanism may be entirely of the student’s own design, or it may incorporate off-the-shelf parts. If the mechanism incorporates off-the-shelf parts, the students must perform appropriate analysis to show that the parts are suitable for their intended purpose in the mechanism.
3. The project must be open-ended – meaning that there is not a known correct answer to the design problem. Students are expected to apply their creativity (simply copying or re-creating something that already exists is not acceptable).
4. The project must have an experimental component. Students must conceive, design, implement and operate an appropriate experiment as part of the project. The experiment might be to collect data about some aspect of the design (i.e., to verify that the design will work as expected). Alternatively, the experiment could be to verify that the final mechanism performs as expected.
5. Upon receiving the approval, the Student Project Group will prepare a preliminary project report consisting Requirement Definition Document, Feasibility Study Document, System Requirement Specification, System Analysis Document, Preliminary System Design Document. All the documents indicated will have a prescribed format.
6. The Project Work will be assessed jointly by a panel of examiners having more than Five Years experience. The Project Groups will deliver the presentation of the Project Work which will be assessed by the panel.
7. The Student Project Group needs to actively participate in the presentation. The panel of examiners will evaluate the candidate’s performance based on presentation skills, questions based on the Project Work, understanding of the Project, analysis and design performed for the project.

8. The Student Project Groups are expected to work on the recommendations given by the panel of examiners. In no case any variation in Project Theme will be permitted.

Assessment Scheme

<table>
<thead>
<tr>
<th>Sr. No.</th>
<th>Content</th>
<th>Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>System Requirement Specification</td>
<td>20</td>
</tr>
<tr>
<td>2</td>
<td>Feasibility Study</td>
<td>10</td>
</tr>
<tr>
<td>3</td>
<td>System Analysis</td>
<td>30</td>
</tr>
<tr>
<td>4</td>
<td>System Design</td>
<td>30</td>
</tr>
<tr>
<td>5</td>
<td>Presentation of the Project Work</td>
<td>10</td>
</tr>
</tbody>
</table>

Note:
The student needs to identify a technological problem in the area of Computer Engineering or Information Technology of their choice and address the problem by formulating a solution for the identified problem. The project work needs to be undertaken by a group of maximum FOUR and minimum of THREE students. The Project work will be jointly performed by the project team members. The Project Group will prepare a synopsis of the project work which will be approved by the concerned faculty member. The project should not be a reengineering or reverse engineering project. In some cases, reverse engineering projects will be permissible based on the research component involved in it. The project work aims at solving a real world technical problem. Hence ample literature survey is required to be done by the students. Application-oriented projects will not be acceptable. Low-level custom User Interface development and its allied mapping with a particular technology will not be accepted. Following is the list of recommended domains for Project Work:

Course Outcomes:
Upon completion of the course, graduates will be able to -

1. Model the Real World Problem
2. Identify the Design within Specification and Available Resources
3. Realise the Solution within Defined references
4. Defend his Design with Technical and Ethical reasoning
5. Adapt to changing Technological and Human resource advances
6. Use the gained knowledge for other Real World Problems