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

Structure & Syllabus of
B.Tech. (Information Technology)

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

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

Signed by

Chairman – BOS        Chairman – Academic Board
## Contents

<table>
<thead>
<tr>
<th>Sr. No.</th>
<th>Title</th>
<th>Page No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Program Educational Objectives &amp; Program Outcomes of B.Tech (Information Technology)</td>
<td>8</td>
</tr>
<tr>
<td>2</td>
<td>Course Structure - Module I and II</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td>2.1 CS10102 Computer programming (Theory Course)</td>
<td>18</td>
</tr>
<tr>
<td></td>
<td>2.2 CS10302 Computer Programming (Laboratory)</td>
<td>21</td>
</tr>
<tr>
<td>3</td>
<td>Course Structure - Module III</td>
<td>23</td>
</tr>
<tr>
<td>4</td>
<td>Course Syllabi for Courses - Module III</td>
<td>24</td>
</tr>
<tr>
<td></td>
<td>4.1 CS20117 Discrete Structures and Graph Theory (Theory Course)</td>
<td>26</td>
</tr>
<tr>
<td></td>
<td>4.2 CS20111 Data Structures (Theory Course)</td>
<td>29</td>
</tr>
<tr>
<td></td>
<td>4.3 CS20108 Computer Organization (Theory Course)</td>
<td>32</td>
</tr>
<tr>
<td></td>
<td>4.4 CS21109 Applied Electronics (Theory Course)</td>
<td>34</td>
</tr>
<tr>
<td></td>
<td>4.5 CS20113 Digital Electronics and Logic Design (Theory Course)</td>
<td>37</td>
</tr>
<tr>
<td></td>
<td>4.6 CS20115 Discrete Structures (Tutorial)</td>
<td>39</td>
</tr>
<tr>
<td></td>
<td>4.7 CS20113 Digital Electronics and Logic Design (Tutorial)</td>
<td>41</td>
</tr>
<tr>
<td></td>
<td>4.8 CS20311 Data structures (Laboratory Course)</td>
<td>42</td>
</tr>
<tr>
<td></td>
<td>4.9 CS20313 Digital Electronics (Laboratory Course)</td>
<td>43</td>
</tr>
<tr>
<td></td>
<td>4.10 CS27401 Mini Project</td>
<td>45</td>
</tr>
<tr>
<td></td>
<td>4.11 CS24301 VB.NET OR</td>
<td>46</td>
</tr>
<tr>
<td></td>
<td>CS24303 C#.NET (Skill Development Course)</td>
<td>48</td>
</tr>
<tr>
<td></td>
<td>4.12 Elective –Soft Skills</td>
<td>--</td>
</tr>
<tr>
<td></td>
<td>4.13 CS20401 Comprehensive Viva Voce</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>4.14 Institute Elective</td>
<td>-</td>
</tr>
<tr>
<td>5</td>
<td>Course Structure - Module IV</td>
<td>50</td>
</tr>
<tr>
<td>6</td>
<td>Course Syllabi for Courses - Module IV</td>
<td>51</td>
</tr>
</tbody>
</table>
6.1 CS21104 Mathematical Transformations and Applications (Theory Course) 53
6.2 CS20110 Computer Graphics (Theory Course) 56
6.3 CS20105 Principles of Programming Languages (Theory Course) 59
6.4 CS21112 Data Communication (Theory Course) 61
6.5 CS20114 Microprocessor and Interfacing (Theory Course) 65
6.6 CS20114 Microprocessor and Interfacing (Tutorial) 67
6.7 CS21104 Mathematical Transforms and Applications (Tutorial) 70
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 Python (Skill Development Course) 78
6.13 Elective – Health & Hobby -
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 CS30117 Human Computer Interaction (Theory Course) 96
8.5 CS30109 Web Technologies (Theory Course) 99
8.6 CS30101 Operating Systems (Tutorial) 102
8.7 CS30117 Human Computer Interaction (Tutorial) 104
8.8 CS30303 Operating Systems (Laboratory Course) 105
8.9 CS30316 Computer Networks (Laboratory Course) 107
8.10 CS30309 Web Technologies (Laboratory Course) 109
## Course Structure - Module VI

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Title</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.1</td>
<td>CS30102 Software Engineering (Theory Course)</td>
<td>118</td>
</tr>
<tr>
<td>10.2</td>
<td>CS30106 Database Management Systems (Theory Course)</td>
<td>121</td>
</tr>
<tr>
<td>10.3</td>
<td>CS30118 Advanced Data Structures and Algorithms (Theory Course)</td>
<td>124</td>
</tr>
<tr>
<td>10.4</td>
<td>CS30114 Systems Programming (Theory Course)</td>
<td>127</td>
</tr>
<tr>
<td>10.5</td>
<td>CS31119 Object Oriented Modeling and Design (Theory Course)</td>
<td>130</td>
</tr>
<tr>
<td>10.6</td>
<td>CS30102 Software Engineering (Tutorial)</td>
<td>133</td>
</tr>
<tr>
<td>10.7</td>
<td>CS30118 Advanced Data Structures and Algorithms (Tutorial)</td>
<td>135</td>
</tr>
<tr>
<td>10.8</td>
<td>CS30314 System programming (Laboratory Course)</td>
<td>137</td>
</tr>
<tr>
<td>10.9</td>
<td>CS30306 Database Management Systems (Laboratory Course)</td>
<td>139</td>
</tr>
<tr>
<td>10.10</td>
<td>CS31139 Object Oriented Modeling and Design (Laboratory Course)</td>
<td>141</td>
</tr>
<tr>
<td>10.11</td>
<td>CS37402 Mini Project</td>
<td>143</td>
</tr>
<tr>
<td>10.12</td>
<td>Professional Development Course (Institute Level)</td>
<td>-</td>
</tr>
<tr>
<td>10.13</td>
<td>CS30402 Comprehensive Viva Voce</td>
<td>-</td>
</tr>
<tr>
<td>10.14</td>
<td>CS37301 Seminar</td>
<td>144</td>
</tr>
</tbody>
</table>

## Course Structure - Module VII

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Title</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1</td>
<td>CS40115 Data Acquisition Systems (Theory Course)</td>
<td>148</td>
</tr>
<tr>
<td>12.2</td>
<td>CS40114 Business Intelligence and Analytics (Theory Course)</td>
<td>150</td>
</tr>
<tr>
<td>12.3</td>
<td>*Elective Group I (Theory Course)</td>
<td></td>
</tr>
<tr>
<td>Course Code</td>
<td>Course Title</td>
<td>Page</td>
</tr>
<tr>
<td>-------------</td>
<td>------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>CS42105</td>
<td>Mobile Computing</td>
<td>155</td>
</tr>
<tr>
<td>CS42127</td>
<td>Cloud Computing</td>
<td>157</td>
</tr>
<tr>
<td>CS42103</td>
<td>Parallel Computing on GPU</td>
<td>160</td>
</tr>
<tr>
<td>CS42101</td>
<td>Advanced Computer Graphics</td>
<td>163</td>
</tr>
<tr>
<td>CS42131</td>
<td>Enterprise Systems</td>
<td>165</td>
</tr>
<tr>
<td>CS42135</td>
<td>Digital Image Processing</td>
<td>168</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>12.4</th>
<th>*Elective Group II (Theory Course)</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>CS42125</td>
<td>Randomized and Approximation Algorithms</td>
<td>170</td>
</tr>
<tr>
<td>CS42136</td>
<td>Distributed Computing</td>
<td>173</td>
</tr>
<tr>
<td>CS42119</td>
<td>Information Retrieval</td>
<td>176</td>
</tr>
<tr>
<td>CS42113</td>
<td>Digital Signal Processing</td>
<td>178</td>
</tr>
</tbody>
</table>

| 12.5        | CS40315 Data Acquisition Systems (Laboratory Course) | 181  |

<table>
<thead>
<tr>
<th>12.6</th>
<th>*Elective Group II (Tutorial Course)</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>CS42125</td>
<td>Randomized and Approximation Algorithms</td>
<td>182</td>
</tr>
<tr>
<td>CS42136</td>
<td>Distributed Computing</td>
<td>184</td>
</tr>
<tr>
<td>CS42119</td>
<td>Information Retrieval</td>
<td>186</td>
</tr>
<tr>
<td>CS42113</td>
<td>Digital Signal Processing</td>
<td>187</td>
</tr>
</tbody>
</table>

| 12.7        | CS40314 Business Intelligence and Analytics (Tutorial Course) | 189  |

<table>
<thead>
<tr>
<th>12.8</th>
<th>*Elective Group I (Laboratory Course)</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>CS42305</td>
<td>Mobile Computing</td>
<td>191</td>
</tr>
<tr>
<td>CS42327</td>
<td>Cloud Computing</td>
<td>193</td>
</tr>
<tr>
<td>CS42303</td>
<td>Parallel Computing on GPU</td>
<td>194</td>
</tr>
<tr>
<td>CS42301</td>
<td>Advanced Computer Graphics</td>
<td>195</td>
</tr>
<tr>
<td>CS42331</td>
<td>Enterprise Systems</td>
<td>197</td>
</tr>
<tr>
<td>CS42335</td>
<td>Digital Image Processing</td>
<td>199</td>
</tr>
</tbody>
</table>

| 12.9        | CS47303 Project Stage - II                      | 200  |

| 13          | Course Structure - Module VIII                 | 203  |

<table>
<thead>
<tr>
<th>14</th>
<th>Course Syllabi for Courses - Module VIII</th>
<th>204</th>
</tr>
</thead>
<tbody>
<tr>
<td>14.1</td>
<td>CS40112 Software Testing and Quality Assurance (Theory Course)</td>
<td>205</td>
</tr>
<tr>
<td>14.2</td>
<td>CS40114 Information Systems Security (Theory Course)</td>
<td>208</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>14.3</th>
<th>*Elective Group III (Theory Course)</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>CS42120</td>
<td>Data Mining</td>
<td>211</td>
</tr>
<tr>
<td>CS42128</td>
<td>Machine Learning</td>
<td>214</td>
</tr>
<tr>
<td>CS42104</td>
<td>Neural Networks</td>
<td>217</td>
</tr>
</tbody>
</table>
### Course Syllabi for PD Courses in TY B.Tech (Information Technology)

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Name</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS33303</td>
<td>Advanced Java</td>
<td>259</td>
</tr>
<tr>
<td>CS33312</td>
<td>PIC Microcontroller</td>
<td>261</td>
</tr>
<tr>
<td>CS33313</td>
<td>Mobile Application Development</td>
<td>263</td>
</tr>
<tr>
<td>CS33306</td>
<td>Ethical Hacking and Network Defense</td>
<td>265</td>
</tr>
<tr>
<td>CS33310</td>
<td>Spring Framework</td>
<td>267</td>
</tr>
<tr>
<td>CS33314</td>
<td>Struts Framework</td>
<td>269</td>
</tr>
<tr>
<td>CS33311</td>
<td>Problem Solving and Programming</td>
<td>271</td>
</tr>
<tr>
<td>CS33315</td>
<td>Big Data Technologies</td>
<td>273</td>
</tr>
<tr>
<td>CS33307</td>
<td>Matlab</td>
<td>275</td>
</tr>
</tbody>
</table>
Program Educational Objectives (PEO)  
B.Tech (Information Technology)

List of Programme Education Outcomes [PEO]

<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 with a commitment towards meeting the needs of users within an organizational and societal context through the selection, creation, application, integration and administration of Information Technology projects.</td>
</tr>
<tr>
<td>PEO2</td>
<td>Core competence</td>
<td>To facilitate students with foundation of mathematical &amp; engineering fundamentals along with knowledge of Information Technology principles and applications and be able to integrate this knowledge in a variety of business and inter-disciplinary setting.</td>
</tr>
<tr>
<td>PEO3</td>
<td>Breadth</td>
<td>To enable student to exercise problem solving capacity with effective use of analysis, design, development that address idea realization.</td>
</tr>
<tr>
<td>PEO4</td>
<td>Professionalism</td>
<td>To inculcate students with professional and ethical values communication and collaboration skill and involvement in team work as a member having multidisciplinary knowledge useful to the society.</td>
</tr>
<tr>
<td>PEO5</td>
<td>Learning Environment</td>
<td>To provide students an academic environment that developed leadership qualities, excellent in subject area of computer engineering and lifelong learning in every sphere of their life.</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>
</table>
| **P01** | GA: 1 Engineering Knowledge | 1. To apply scientific and mathematical principles in order to determine conceptual aspects of real world problems in information engineering.  
2. To apply algorithmic principles and information science theory for comprehending technological trade-off.  
3. To explore conceptual paradigms with incorporation of programming practices. |
| **P02** | GA: 2 Problem Analysis | 4. To recognize and synthesize the context of the problem leading to correct and consistent requirements.  
5. To analyze and formulate problem frames in order to receive decomposition structure of information engineering/technology problem.  
6. To identify resources, infrastructure and technology required to realize solution of real world problem. |
| **P03** | GA: 3 Design/Development of solution | 7. To plan and devise design alternatives which leads to conceive optimal solution.  
8. To compose technical design specifications for formally expressing the solution implementation.  
9. To practice template based approaches for formulating engineering artifacts addressing information engineering/technology problem. |
| **P04** | GA: 4 Conduct Investigation of Complex Problem | 10. To apply research knowledge in order to recognize information engineering/technology problem issues.  
11. To investigate real world information engineering/technology problem with cause effect analysis and inference.  
12. To apply research methods to determine impact and severity of |
| PO5          | Modern Tool Usage | 13. To select appropriate tools for solution development.  
|             |                  | 14. To demonstrate ability to formulate and answer empirical questions.  
|             |                  | 15. To apply techniques and methods to create, enhance, and deliver IT tools.  
| PO6          | The Engineer and Society | 16. To devise engineering solutions in a meaningful and useful way to address societal needs.  
|             |                  | 17. To impart technological solutions with legal commitments and cultural diversity.  
| PO7          | Environment and sustainability | 18. To recognize impact of engineering solutions on society.  
|             |                  | 19. To adapt to changing technological scenarios in order to realize socio-technical solutions.  
| PO8          | Ethics | 20. To adhere to ethical responsibility.  
|             |      | 21. To follow norms of engineering practice.  
| PO9          | Individual and Team Work | 22. To interact professionally at work places with effective team work.  
|             |      | 23. To demonstrate synergistic leadership skills while addressing multi-disciplinary complex problems.  
| PO10         | Communication | 24. To demonstrate proficiency in technical and social communications.  
|             |      | 25. To interpret and represent engineering artifacts considering IT environment.  
| PO11         | Lifelong Learning | 26. To undertake refresher courses and consultancy projects with participation in continuous development of organization.  
|             |      | 27. To improve the skills in refining and updating information engineering knowledge base.  
|             |      | 28. To strive for continuous career building in information technology by higher education.  
| PO12         | Project Management | 29. To determine project specific constraints, forces, resources and schedule.  
|             |      | 30. To acquire projects from competitive global world by
| and Finance | conceptualizing and framing unique ideas. |
## 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>TOTAL</td>
<td></td>
<td></td>
<td>14</td>
<td>8</td>
<td>2</td>
</tr>
</tbody>
</table>

<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>HS10102</td>
<td>Differential Integral and Calculus</td>
<td>Theory - Core</td>
<td>3</td>
<td>-</td>
<td>1</td>
</tr>
<tr>
<td>HS10104</td>
<td>Engineering Mechanics</td>
<td>Theory – Core</td>
<td>3</td>
<td>-</td>
<td>1</td>
</tr>
<tr>
<td>HS10108</td>
<td>Electrical Engineering Fundamentals</td>
<td>Theory – Core</td>
<td>3</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>CS10102</td>
<td>Computer Programming</td>
<td>Theory – Core</td>
<td>3</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>HS16102</td>
<td>Economics Management Technology</td>
<td>Theory – OE</td>
<td>2</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>HS16104</td>
<td>Cost &amp; Acc. Business Law</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HS16106</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CS10302</td>
<td>Computer Programming</td>
<td>Lab – Core</td>
<td>-</td>
<td>2</td>
<td>-</td>
</tr>
<tr>
<td>HS10306</td>
<td>Engineering Lab</td>
<td>Lab – Core</td>
<td>-</td>
<td>2</td>
<td>-</td>
</tr>
<tr>
<td>HS17402</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>14</td>
<td>8</td>
<td>2</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</td>
<td>Tut.</td>
<td>ISA</td>
</tr>
<tr>
<td>HS10107</td>
<td>Communication Skill</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>HS15301</td>
<td>English I</td>
<td>Language</td>
<td>-</td>
<td>2</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>-</td>
<td>2</td>
<td>-</td>
</tr>
<tr>
<td>TOTAL</td>
<td></td>
<td></td>
<td>-</td>
<td>8</td>
<td>-</td>
</tr>
</tbody>
</table>

### 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></td>
<td></td>
<td></td>
<td>Test 1</td>
<td>Test 2</td>
<td>HA</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 -

**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.


Unit 2: 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.

Page 16 of 251
Unit 4: Pointers and Strings  
**PART A:** Concept of pointers, relevance of data type in pointer variable, pointer arithmetic. Pointer to pointer. Pointers and functions (passing pointers to functions, returning pointers from functions). Pointers and arrays. Pointers and strings. Pointer constants. Array of pointers, pointer to array. Various alternatives of accessing arrays (1-D and 2-D) using pointers. 

**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  
**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.

**PART B:** Typedef keyword. Union, Nesting of Structure and Union. Enumerated data types.

**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 sting.
   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 sting.
   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
# Course Structure

**Branch** – Computer Engineering  
**Year** - THIRD Year of Engineering  
**Semester** - 1  
**Academic Year** – 2015-2016  
**Pattern** – f11

## Theory Courses

<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>CS30101</td>
<td>Operating Systems</td>
<td>3</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>CS30116</td>
<td>Computer Networks</td>
<td>3</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>CS30105</td>
<td>Theory of Computation</td>
<td>3</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>CS30117</td>
<td>Human Computer Interaction</td>
<td>3</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>CS30109</td>
<td>Web Technologies</td>
<td>2</td>
<td>0</td>
<td>5</td>
</tr>
</tbody>
</table>

**TOTAL**  
14  2  16

## Laboratory Courses

<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>Practicals</td>
<td>ISE</td>
<td>ESE</td>
</tr>
<tr>
<td>CS30303</td>
<td>Operating Systems</td>
<td>2</td>
<td>30</td>
<td>70</td>
</tr>
<tr>
<td>CS30316</td>
<td>Computer Networks</td>
<td>2</td>
<td>30</td>
<td>70</td>
</tr>
<tr>
<td>CS30309</td>
<td>Web Technologies</td>
<td>2</td>
<td>30</td>
<td>70</td>
</tr>
<tr>
<td>CS37401</td>
<td>Mini Project(T.Y. Semester I)</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PD1</td>
<td>Institute Level Elective</td>
<td>2</td>
<td>30</td>
<td>70</td>
</tr>
<tr>
<td>CS30401</td>
<td>Comprehensive Viva Voce</td>
<td>Based on Courses S1, S2,</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>CS37301</td>
<td>Seminar(T.Y. Semester I)</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Irrespective of Module</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------------------------</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td></td>
</tr>
<tr>
<td>CS37302</td>
<td>PROJECT STAGE 1</td>
<td>4</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>(T.Y. Semester II)</td>
<td>Irrespective of Module</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TOTAL</td>
<td>20</td>
<td></td>
<td>12</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Abbreviations Used</th>
<th>TOTAL CREDITS</th>
</tr>
</thead>
<tbody>
<tr>
<td>HA</td>
<td>Home Assignment</td>
</tr>
<tr>
<td>TA</td>
<td>Teacher Assessment</td>
</tr>
<tr>
<td>ISE</td>
<td>In Sem. Examination</td>
</tr>
<tr>
<td>ESE</td>
<td>End Sem. Examination</td>
</tr>
<tr>
<td>CA</td>
<td>Continuous Assessment</td>
</tr>
<tr>
<td>*</td>
<td>Alternate Week Labotatory</td>
</tr>
<tr>
<td>^</td>
<td>Detail Enclosed Separately</td>
</tr>
</tbody>
</table>
CS30101: OPERATING SYSTEMS

Credits: 03

Teaching Scheme: - Theory 3 Hrs/Week

Prerequisites: Data Structures and Algorithms, Computer Organization.

Unit 1: Introduction to OS

PART A: Introduction to OS: What is OS, Interaction of OS and hardware, Goals of OS, Basic functions of OS, OS Services, System Calls.

Types of OS: Batch, Multiprogramming, Time sharing, Parallel, Distributed & Real-time OS.

Structures of OS: Monolithic, Layered, Ringed, Virtualization-Virtual Machines, Hypervisor, Exokernels, Client-server model, Microkernels.

Shell: Linux commands and shells, shell programming, AWK programming.

Introduction to Mobile OS: Architecture & Overview of Android OS.

PART B: Overview of Linux and Windows 2000 architecture

Unit 2: Process Management


Threads: Multithreading models, Thread implementations – user level and kernel level threads.

Symmetric Multiprocessing.

Concurrency: Issues with concurrency, Principles of Concurrency


Unit 3: Scheduling and Deadlock


Scheduling Algorithms: FCFS, SJF, RR, Virtual Round Robin, Priority

Multiprocessor Scheduling: Granularity, Design Issues, Process Scheduling


PART B: Thread Scheduling, Real Time Scheduling.

Unit 4: Memory Management

PART A: Memory Management concepts: Memory Management requirements, Memory Partitioning: Fixed, Dynamic Partitioning, Buddy Systems. Placement Strategies: First Fit, Best Fit, and Worst Fit, Fragmentation, Swapping, Paging,

Segmentation, Address translation.

**Virtual Memory**: Concepts, VM with Paging, Page Table Structure, Inverted Page Table, Translation Lookaside Buffer, VM with Segmentation.

**OS policies for Virtual Memory**: Fetch, Placement, Replacement, Resident Set management, Cleaning Policy, Load Control.


**PART B**: VM with combined paging and segmentation, Working Set Model.

**Unit 5**: I/O and File Management (8+1 Hrs)

**PART A**: I/O management: I/O Devices - Types, Characteristics of Serial and Parallel devices, OS design issues for I/O management, I/O Buffering.

**Disk Scheduling**: FCFS, SCAN, C-SCAN, SSTF.


**File System**: Structure, Implementation, Memory mapped files, Special Purpose File Systems

**Case study**: Process Management, Concurrency, Scheduling, Memory Management, I/O Management, File Management(VFS) in LINUX Shell and Command Programming, AWK Programming.

**PART B**: Organization of I/O functions, Disk Caches.

**Text Books**


**Reference Books**


**Additional Reading**

Course Outcomes:

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

1. Utilize mechanisms and strategies of an Operating System to solve real world problems which have similar requirements as the various core functionalities of an Operating System.
2. Explain the issues involved in design and implementation of operating systems in various contexts.
3. Analyze the tradeoffs inherent in operating system design.
4. Identify the objectives and functions of modern operating systems.
5. Examine the functions of a contemporary operating system with respect to convenience, efficiency and the ability to evolve.
CS30116:: COMPUTER NETWORKS

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

Prerequisites: Data Communication.

Unit 1: Introduction to Computer Networks and Logical Link Control (8+1 Hrs)
PART B: Point-to-Point Protocol (PPP), MPLS, Bridges, Gateways, Network Cables

Unit 2: Medium Access Control (8+1 Hrs)
PART A: Channel allocation: Static and Dynamic allocation, Multiple Access Protocols: Pure ALOHA, Slotted ALOHA, CSMA, WDMA, Ethernet: Cabling, MAC Sub-layer protocol: DIX and IEEE 802.3 Frame Formats, Collision Detection, Binary Exponential Back-off Algorithm, Switched Ethernet, Fast Ethernet, Wireless 802.11a/b/g/n LANS, MACA, Broadband wireless: 802.16
PART B: Gigabit Ethernet, Layer-II Switch and Bluetooth

Unit 3: Network Layer (9+2 Hrs)
PART B: Broadcast and Multicast routing, Routing for mobile hosts, IGMP, Mobile IP, VLAN

Unit 4: Transport Layer (8+1 Hrs)

PART B: Real Time Streaming Protocol RTSP, RTP, RTCP

Unit 5: Application Layer (7+1 Hrs)

PART A: Domain Name System (DNS), Naming and Address Schemes, DNS servers, E-mail: MIME, SMTP and POP3. Remote login, File Transfer Protocol (FTP), SNMP, DHCP and BOOTP. CDN, Working of Bit Torrent, Cloud computing: Architectures and working principle.


Text Books

Reference Books

Additional Reading

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

1. Interpret topological network architectures and essential components to design it.
2. Estimate reliability issues based on error control, flow control and pipelining by using bandwidth, latency, throughput and efficiency.
3. Uniformly demonstrate LAN behavior utilizing network architecture, protocols, and network components.
4. Design client server based applications using sockets.
5. Demonstrate data flow between peer to peer in an IP network using Application, Transport and Network Layer Protocols.
CS30105:: THEORY OF COMPUTATION

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

Prerequisites: Data Structures.

Unit 1 : Automata Theory  
PART B: FA with output: Moore and Mealy machine.

Unit 2 : Regular Expressions (RE) and Languages  
PART A: Regular expression (RE), Definition, Operators of regular expression and their precedence, Algebraic laws for Regular expressions, Kleen's Theorem, Regular expression to DFA, DFA to Regular expression, Non Regular Languages, Pumping Lemma for regular Languages, Myhill-Nerode theorem, Closure properties of Regular Languages, Applications of RE: Regular expressions in Unix, GREP utilities of Unix, Lexical analysis and finding patterns in text.
PART B: Decision properties of Regular Languages.

Unit 3 : Context Free Grammars (CFG) and Push Down (11+1 Hrs) Automata (PDA)  
PART A: Context Free Grammars: Definition, Examples, Derivation, Languages of Grammar, Derivation trees, Ambiguity in Grammar, Ambiguous and Unambiguous CFG, Inherent ambiguity, Simplification of CFGs, Normal forms for CFGs: CNF and GNF, Closure properties of CFLs, Decision Properties of CFLs (Emptiness, Finiteness and Membership), Chomsky Hierarchy, Pumping lemma for CFLs
Push Down Automata: Description and definition, Language of PDA, Acceptance by Final state, Acceptance by empty stack, Deterministic PDA, CFG to PDA construction (with proof), Equivalence of PDA and CFG (without proof).
PART B: Regular grammars, left linear and right linear regular grammars, regular grammar and finite automata.

Unit 4 : Introduction to Turing Machines  
PART A: Turing Machines: Basic model, definition and representation, Instantaneous Description, Language acceptance by TM. Robustness of Turing Machine model and equivalence with various variants: Two-
way/One-way infinite tape TM, multi-tape TM, non-deterministic TM, TM as enumerator.

Recursive and Recursively Enumerable languages and their closure properties.

PART B : Comparison between Finite Automata, Push Down Automata, and Turing Machines.

Unit 5: Introduction to Undecidability (6+1 Hrs)

PART A : Universal Turing Machines, Church-Turing Thesis and intuitive notion of Algorithm.

Introduction to countable and uncountable sets (countability of set of natural numbers, integers, rationals. Uncountability of set of real numbers, points in plane), Encoding for Turing machines and countability of set of all Turing machines. Existence of Turing unrecognizable languages. Undecidability of Halting problem, Post Correspondence Problem. Example of a Turing unrecognizable language. Decision properties of R, RE languages and Rice’s theorem.

PART B: Hilbert’s tenth problem, undecidability of tiling problem

Text Books

Reference Books

Additional Reading

Course Outcomes:

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

1. To infer the applicability of various automata theoretic models for recognizing formal languages.
2. To discriminate the expressive powers of various automata theoretic and formal language theoretic computational models.
3. To illustrate significance of non determinism pertaining to expressive powers of various automata theoretic models.
4. To comprehend general purpose powers and computability issues related to
state machines and grammars.

5. To explain the relevance of Church-Turing thesis, and the computational equivalence of Turing machine model with the general purpose computers.

6. To grasp the theoretical limit of computation (independent of software or hardware used) via the concept of undecidability.
CS30117:: HUMAN COMPUTER INTERACTION

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

Prerequisites:

Unit 1 : Introduction  (7+1 Hrs)
P A R T B: Identification of Application Category and Related Features for Selected Product / System.

Unit 2 : Principles, Models & Guidelines  (9+2 Hrs)
P A R T B: Task / Error Analysis for Selected Product / System.

Unit 3 : Design Process and Interaction Styles  (8+2 Hrs)
P A R T B: UI Design for Selected Product/System.

Unit 4 : Evaluation Techniques and Interface Categories  (8+2 Hrs)
P A R T B: Usability Evaluation of Selected Product/System.

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

PART 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
CS30109 :: WEB TECHNOLOGIES

Credits: 02
Teaching Scheme: 2 Hrs/Week

Prerequisites: Principles of Programming Languages.

Unit 1 : Introduction to Web Technologies (6+1 Hrs)
PART A: World wide web, what we mean by "web technologies", Evolution of web: front content to social to semantic (Web 1.0, 2.0, 3.0, web as a platform) Introduction to Client side technologies: HTML, CSS, JavaScript, AJAX.
HTML : HTML Elements, Attributes, HTML formatting Tags for skeleton, text, links, lists, images, styles, Tables, Frames, Introduction to CSS,
JavaScript : Data Types and Variables, Operators and expressions, Arrays, Loops, conditions, JavaScript Objects, Forms & Reg Expressions, Events, JavaScript Browser Functions.
PART B: Applications of JavaScript, Validations Using JavaScript, Creating Menu using CSS.

Unit 2 : JQuery and Data formats and representations – I (5+1 Hrs)
HTML DOM : Predefined Objects, Object Hierarchy, Accessing Objects, Event Handlers, Node : Create, Add, Insert, Clone, Remove and Replace.
PART B : Creating Animations and Slider using JQuery, Referring DOM using JavaScript.

Unit 3 : Data formats and representations – II (5+1 Hrs)
PART A: XML: Role of XML, Prolog, Elements, Attributes, Namespace XML DTD : Purpose of DTD, Using DTD in XML, Element Type Declaration, Attribute Declaration XSD : Limitations of DTD, Schema Elements, Element Definition, Schema Schema Validation, Built in Data Types. The Java API for XML parsing (JAXP), X-Path, Overview of XSLT.
PART B: X-query.

Unit 4 : Server Side Technologies (6+1 Hrs)
JSP : JSP lifecycle, Directives, Comments, Expressions, Scriptlets, Declaration, Scope of JSP Objects, Standard Actions, Introduction to JavaBeans, Calling JavaBean from JSP Page. Client side security : Cookies security policy , Server side security tools e.g. Web Application Firewalls (WAFs) and fuzzers.
PART B: Session and Cookies management in Servlet and JSP.
Unit 5: Distribution of data  

(5+1 Hrs)


PART B: AJAX: AJAX Benefits, Basic Idea, Asynchronous Communication, AJAX Processing Steps,

Text Books


Reference Books


Course Outcomes:

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

1. Design static and dynamic web pages using HTML, PHP and XML Technologies.
2. Apply the effects of CSS, Javascript, JQuery and XSLT in Web design.
4. Evaluate problems and alternative web solutions using Web Technologies in a wide variety of business and organizational contexts.
5. Create data-driven web applications using JSP and Servlet.
6. Incorporate best practices for building applications with Struts.
CS30101:: OPERATING SYSTEMS

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

Prerequisites:: Data Structures and Algorithms, Computer Organization.

List of Contents

A TERM-WORK containing the record of the following:

1. Execution of Advance Unix commands.

2. Write a shell program to sort an array of numbers using any sort method.

3. Execution of AWK related commands.

4. Implement the solution for Reader-Writer problem using Threads and Semaphores/Mutex.

5. Implement the solution for Producer-Consumer (Bounded Buffer) problem using Threads and Semaphore/Mutex.

6. Implement the solution for Dining-Philosopher problem using Threads and Semaphore.

7. Implementation of resource allocation graph (RAG).

8. Implement the solution for Banker’s Algorithm for deadlock avoidance.

9. Draw the Gantt charts and compute the finish time, turnaround time and waiting time for the following algorithms:
   a. First come First serve
   b. Shortest Job First (Preemptive and Non-Preemptive)
   c. Priority (Preemptive and Non-Preemptive)
   d. Round Robin

10. Calculate the number of page faults for a reference string for the following page replacement algorithms:
    a. Optimal
    b. FIFO
    c. LRU
11. Calculate the total distance traversed by the disk arm to satisfy the pending requests for the following disk scheduling algorithms:
   a. FCFS
   b. SSTF
   c. SCAN
   d. C-SCAN

**Text Books**

**Reference Books**

**Additional Reading:**
CS30117:: 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. Design user persona for the users of selected product/system.
2. Perform GOMS analysis for 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 testing / evaluation technique.

Text Books

Reference Books

Additional Reading
CS30303:: OPERATING SYSTEMS

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

Prerequisites: Data Structures and Algorithms, Computer Organization

List of Practical
Part A:

1. Implementation of a multiprogramming operating system:
   a. Stage I:
      i. CPU/ Machine Simulation
      ii. Supervisor Call through interrupt
   b. Stage II:
      i. Paging
      ii. Error Handling
      iii. Interrupt Generation and Servicing
      iv. Process Data Structure
   c. Stage III:
      i. Multiprogramming
      ii. Virtual Memory
      iii. Process Scheduling and Synchronization
      iv. Inter-Process Communication
      v. I/O Handling, Spooling and Buffering

Text Books

Reference Books

Additional Reading
CS30316:: Computer Networks

Credits: 01

Teaching Scheme: - Laboratory 2 Hrs/Week

Prerequisites: Data Communication

List of Practical

1. Set up a small network of 2 to 4 computers using Hub/Switch. It includes installation of LAN Cards, Preparation of Cables, Assigning IP addresses and sharing C drive.

2. File Transfer using PC To PC Communication.


4. Studying Linux and Windows network commands. [ ping, pathping, ipconfig/ifconfig, arp, netstat, nbtstat, nslookup, route, traceroute/tracert, nmap, etc]

5. Program for calculating CRC using Modulo-2 and Polynomial methods.

6. Simulate the sliding window protocols Go Back N and Selective Repeat.

7. File Transfer between two computers using TCP sockets.

8. Multiuser chat application using UDP sockets.

9. To create TCP/IP packet using standard TCP/IP include files and send it to other machine

10. Program to find active and passive ports on nearby host using sockets.


12. Installing and configuring DHCP server for Linux/Windows.

Text Books

Reference Books

Additional Reading
Prerequisites: Principles of Programming Languages

List of Tutorials
1. Design a Simple web page using HTML.
2. Design an Attractive web page using CSS.
3. Design a web page using HTML and implement validations using Java Script and give effects using JQuery.
4. Design an XML page and there DTD and validate it using Java API for XML parsing.
5. Design an XML page and there XSD and validate it using Java API for XML parsing.
6. Design web page using a Java Servlet to read and write data to database using JDBC.
7. Design an interactive web page using JSP.
8. Design an interactive web page using PHP.

Text Books
4. “XML and Related Technologies” by Atul Kahate, PEARSON Education.

Reference Books
CS37401::MINI PROJECT

Credits: 02

The Student has to select a project in group based on a topic of interest from any of the subjects offered in current Semester. Periodically the implementation will be evaluated by the guide.

Evaluation is done in two stages. In the first review the internal Guide evaluates the project against 40% of the implementation of work. At the end of semester each group will be evaluated by externally Guide from Industry based on their Presentation, completeness of Project implementation and report artifact.

Course Outcomes
Upon completion of the course, graduates will be able to -
1. Recognize essential & dominant area of technology for achievable artifacts over rapid period of time.
2. Acquire rapid application development cycle involving prototyping to learn adequate technological environments.
3. Concisely formulate specific problem in drafted specification format.
4. Devise data dictionaries and solution design with sufficient details.
5. Demonstrate the crafted solutions to user community with a lean learning curve.
6. Validate newer dimension of extendable and scalable nature of the problem solution crafting.
CS37301::SEMINAR

Credits: 02  
Teaching Scheme: - Lab 2 Hrs/Week

Seminar is a course requirement wherein under the guidance of a faculty member a student is expected to do an in-depth study in a specialized area by doing literature survey, understanding different aspects of the problem and arriving at a status report in that area. Students are expected to choose a topic in CSE based on current trends or industry practices. While doing a seminar, the student is expected to learn investigation methodologies, study relevant research papers, correlate work of various authors/researchers critically, study concepts, techniques, prevailing results etc., analyze it and present a seminar report. Evaluation will be based on relevance of topic, understanding of the problem, literature Survey, presentation, communication skills, answering queries and reporting or documenting procedure.

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

1. Scope and visibly identify technological trade off in computer engineering leading to significant topics.
2. Conduct a thorough literature survey of identify technical topic.
3. Present technical topic in written form with technical report or document
4. Communicate effectively technical topic in verbal form with suitable demonstration
5. Access real world problem scenarios in computer engineering.
6. Demonstrate skills and competences with an awareness of technical standardization.
CS37302::PROJECT STAGE I

Credits: 02

Aim
This course addresses the issues associated with the successful management of a 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 working with peers to create desired outcomes. The Project Work can lead to:

a. Transform existing Ideas 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 based on a Fresh Idea or Implementation of a Theoretical Problem – meaning that there is not a known Solution to the design problem Or Create a Better Solution.
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 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. 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.

Assessment Scheme

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

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

1. Identify Real World Problems
2. Apply Computing Solutions to Real World Problems
3. Construct a Solution Model to Real World Problem
4. Select Design Pattern to Best approach the Solution.
5. Lay Down rules to Minimise Adverse Impact of Design Implementation
6. Adapt to changing Technological and Human resource advances.

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:

<table>
<thead>
<tr>
<th>Computer Networks</th>
<th>Image Processing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating Systems</td>
<td>Artificial intelligence</td>
</tr>
<tr>
<td>Network Security</td>
<td>Expert Systems</td>
</tr>
<tr>
<td>Digital Signal Processing</td>
<td>Object Oriented Systems</td>
</tr>
<tr>
<td>Subject</td>
<td>Topic</td>
</tr>
<tr>
<td>----------------------------------------------</td>
<td>-------------------------------</td>
</tr>
<tr>
<td>Systems Programming</td>
<td>Modeling and Design</td>
</tr>
<tr>
<td>Real Time Systems</td>
<td>System Testing</td>
</tr>
<tr>
<td>Embedded systems</td>
<td>Storage Management</td>
</tr>
<tr>
<td>Cluster Computing</td>
<td>Client-Server Computing</td>
</tr>
<tr>
<td>Mobile &amp; Wireless Communications</td>
<td>Cloud Computing</td>
</tr>
<tr>
<td>Multimedia Systems</td>
<td>Protocol Engineering</td>
</tr>
</tbody>
</table>
CS33303:: ADVANCED JAVA

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

Prerequisites: Core Java.

List of Practical

1. Design a java application to demonstrate the use Java revision, anonymous inner classes, file handling, GUI, event handling, debugging using IDE.
2. Design a java application to demonstrate use of Multithreading, concurrency, synchronous and asynchronous callbacks, ThreadPools using ExecutorService.
3. Design a java application to demonstrate use of Collections and generics.
4. Design a java database application using multithreading and concurrency control.
5. Design a java application to demonstrate use of Servlets and JSP.
6. Design a client-server application demonstrating the use of Java I/O using sockets with GUI for configurations.
7. Design a java RMI application.
8. Designing a java application to demonstrate use of Web Services - REST and SOAP.
9. Design a java application to demonstrate dynamic invocation using reflection.

Reference Books


Course Outcomes:

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

1. Select the advanced features of java in solving a complex problem.
2. Implement appropriate exception handling in code.
3. Choose the appropriate advanced java features depending on problem statement.
4. Practice an IDE like Eclipse or Netbeans for quicker coding/debugging.
5. Produce reusable and extensible design to minimise rework.
6. Construct the solution by breaking the complex problem into smaller problems.
CS33312:: PIC Microcontroller

Credits: 01

Teaching Scheme: - Laboratory 2 Hrs/Week

Prerequisites: Microprocessor, x86 , x86 Interfacing Chips.

List of Practical

1. Assignment on Program Compilation and Burning into Microcontroller.
2. Assignment on Input Output.
3. Assignment on Interrupt.
4. Assignment on LED.
5. Assignment on Timer.
6. Assignment on LCD.
7. Assignment on UART.
8. Assignment on Write and Read from EEPROM.
9. Assignment on ADC.
10. Assignment on PWM.
11. Assignment on Stepper Motor.

Text Books

1. Data Sheet  www.microchip.com
2. Hitachi Data Sheet on LCD HD 44780

Reference Books

1. Microchip 18F45xx

Course Outcomes:

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

2. Utilize the Structures to effectively solve Computing Problems.
3. Design system interconnects for effective throughput.
4. Validate design outputs using standards test equipment.
5. Design Effective Automation Solutions.

Cooperate with diverse Teams for delivering automation Solutions.
CS33313:: MOBILE APPLICATION DEVELOPMENT

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

Prerequisites: Java.

List of Practical

1. Download, Install and Configure Eclipse IDE with Android Development Tools (ADT) plug-ins and Android SDK or Android Studio or Net Beans with Android plugin.

2. Building Simple User Interface using UI Widgets such as Buttons, Text Fields and View.

3. Design an android based application using content provider.

4. Develop an android based application to implement the sequential and random file operation.

5. Develop an android based application to create simple embedded database for the student attendance and find defaulters in the class using SQLite.

6. Design an android based application to demonstrate GPS services using Google map.

7. Design an android based application to implement HTTP operations for internet communication.

8. Design an android based application to implement chat application using socket programming.

9. Design an android based application to take a snapshot by using the Camera in your mobile. Save the snapshot in the image or video format. Use Camera Media API provided Android.

10. Mini Project.

Text Books


Reference Books


Course Outcomes:

Upon completion of the course, graduates will be able to
1. Use embedded database SQLite, Flat files and Multi Media files.
2. Display the current location of a device using google map.
3. Develop the user interface.
4. Choose suitable software tools and APIs for the development of Mobile Application
5. Design and deploy mobile application using software development environment
6. Demonstrate internet based application.
CS33306: ETHICAL HACKING AND NETWORK DEFENSE

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

Prerequisites: Fundamentals of IT, Networking, Microsoft OS, LINUX or UNIX operating systems.

List of Practical
1. Study of different type of attacks
2. Study of Ethical hacking, types of hacking, different phases involved in hacking.
3. Study of skills to become ethical hacker.
4. Study of spoofing techniques
5. Study of password cracking techniques
7. Study of spyware technology
8. Study of types of viruses, antivirus techniques and virus detection mechanism
9. Study of Sniffing techniques and tools.
10. Study of Flooding attacks like MAC flooding, SYN flooding etc.
11. Study of Session Hijacking and prevention of session hijacking.
12. Web based password cracking techniques
14. Study of Physical security.

Text Books
Michael T Simpson – “Ethical Hacking and Network Defense”.

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

1. Analyze nature and type of attack.
2. Establish type of attack on a given system.
3. Simulate different types of attacks using tools.
4. Differentiate between the type of communication services used for attack.

5. Design a secure system for protection from the various attacks 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.
CS33310: SPRING FRAMEWORK

Credits: 01

Teaching Scheme: - Laboratory 2 Hrs/Week

Prerequisites: Java, JSP, Servlets

List of Practical

1. Assignment on Spring Environment Setup.
2. Assignment on Spring Hello World Example.
3. Assignment on Spring IOC Container.
4. Assignment on Spring Bean Scopes, Spring Bean Life Cycle.
5. Assignment on Spring Bean Post Processors.
6. Assignment on Spring Dependency Injection, Spring Injecting Inner Beans, Spring Injecting Collection, Spring Beans Auto-Wiring.
7. Assignment on Spring Annotation Based Configuration, Spring Java Based Configuration.
8. Assignment on Event Handling in Spring.
9. Assignment on Spring AOP Assignments.
10. Assignment on Spring JDBC assignments.
11. Assignment on Spring Web-MVC Assignments.

Text Books

1. Spring Recipes – A problem solution approach by Gary Mak, Josh Long and Daniel Rubio.
2. Professional Java Development with the Spring Framework, by Rod Johnson

Reference Books

1. Pro Spring 3.0 by Clarence Ho, Rob Harrop.
2. Expert Spring MVC and Web Flow by Seth Ladd, Darren Davison, Steven Devijver, Colin Yates

Course Outcomes:

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

2. Construct formalized design patterns to effectively implement Java Enterprise Application lifecycle.
3. Create application using Spring Tool Suite, Software project management and comprehension tool like Maven.
4. Demonstrate that the business rules and validations are implemented in shorter time using this framework.
5. Acquire skills to work on real time projects in industry.
6. Use pre-built framework for rapid application development using Spring Framework MVC Applications.
CS33311: STRUTS FRAMEWORK

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

Prerequisites: Java, JSP, Servlets

List of Practical

1. Building a Simple Struts Application
2. Struts validator framework
3. Setup validator framework in Struts
4. Struts validator Framework
5. Using the validator framework in struts
6. Validator framework work in Struts
7. Sing validator framework work in struts
8. Using the validator Framework
9. Fixed Value check using struts validator framework
10. Struts 2 double validator
11. Struts 2 Date validator
12. Client Side Address Validation in Struts
13. Struts 2 RequiredString validator
14. Struts 2 E-mail Validator
15. XML files used in Validator Framework?
16. struts - Framework
17. Struts 2 Validation (Int Validator)
18. Struts 2 Url Validator
19. Validation using validator-rules.xml – Struts

Text Books

1. "Jakarta Struts Live" by Rick Hightower published by SourceBeat.

Reference Books


Course Outcomes:

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

2. Explain Action Mappings, Forms, JSP Standard Tag Library, Internationalization And Localization, Input Validation and Advanced Configuration.
4. Demonstrate that the business rules and validations are implemented in shorter time using this framework.
5. Acquire skills to work on real-time projects in industry.
6. Incorporate best practices for building applications with Struts
CS33314:: PROBLEM SOLVING AND PROGRAMMING

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

Prerequisites: Data structures.

List of Practical

1. Data structure review (stack, queue, linked list).

2. Graph searching techniques (DFS, BFS, IDDFS etc.) and applications of graph searching in problems in programming competition.

3. Advanced data structures union-find (including optimized algorithms like path compression), segment trees, interval trees, augmented data structures and their applications.

4. String searching algorithms.

5. Dancing links to speed up backtracking.


Text Books


Reference Books


Course Outcomes:

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

1. Apply and practice logical ability to solve the problems.

2. Modularize the problems into small modules and then convert them into
3. Analyze algorithms and determine their time complexity.
4. Trace and code recursive programs.
5. Choose appropriate problem solving technique
6. Verify and validate the correctness of the algorithm.
CS33315:: BIG DATA TECHNOLOGIES

Credits: 01

Teaching Scheme: - Laboratory 2 Hrs/Week

Prerequisites: Programming Skills

List of Practical

1. Study of Hadoop 1 / Hadoop 2 (YARN)
2. Study of hadoop distributed file system (HDFS)
3. Manipulation of data on HDFS
4. Learning Map Reduce Programming
5. Word count problem using Map Reduce Programming
6. Hands-on over Pig
7. Hands-on over Hive
8. Introduction to Hbase

Text Books

2. “Programming Pig”, Allen Gates, O'Reilly

Reference Books

1. “Programming Hive”, Dean Wampler, O’Reilly
2. “HBase: The Definitive Guide”, Lars George, O’Reilly

Additional Reading

1. "Hadoop In Action", Chuck Lam, Manning Publication

Course Outcomes:

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

1. Illustrate architecture of Hadoop
2. Break down a computing problem into multiple parallel tasks
3. Explain Hadoop Ecosystem
4. Organise input data to handle it using HDFS
5. Apply map reduce programming technique to address real world problems
6. Adapt to upcoming technologies for management of complex big data problems
CS33307:: MATLAB

Credits: 01

Teaching Scheme: - Laboratory 2 Hrs/Week

List of Practical (Any Ten)

1. Introduction to MATLAB, MATLAB Elements & Simple Programs and debugging concepts.
2. Write a Matlab Program for functions.
3. Write a Matlab Programs by using IF Then Else, Case, Statement, for Loop, While loop.
4. Write a Matlab Program for 2-D graph.
5. Write a Matlab Program for 3-D graph.
6. Write a Matlab Program for various Image operations.
7. Write a Matlab Program for Animations.
8. Study of MATLAB debugging commands.
9. Write a Matlab Program to create GUI.
10. Write a Matlab Program to simulate a simple circuit.
11. Write a Matlab Program to create Movie.
12. Write MATLAB Program to read sound file and adjust its parameters.
13. Write MATLAB Program to read .avi file.

Text Books


Reference Books

Course Outcomes:

1. Upon completion of the course, graduates will be able to -
2. Solve Mathematical equations.
3. Design GUI by using MATLAB.
5. Validate design outputs using standards test equipments.
6. Develop animation programs by using MATLAB.
7. Perform various operations on Image.