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>3.7 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 CS24303 C#.NET (Skill Development Course)</td>
<td>46</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>
### Course Structure - Module V

| 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 | 78 |
|       | CS24304 | Python (Skill Development Course) | 80 |
| 6.13 |       | Elective – Health & Hobby | - |
| 6.14 | CS20402 | Comprehensive Viva Voce | - |

### Course Syllabi for Courses - Module V

| 8.1 | CS30101 | Operating Systems (Theory Course) | 85 |
| 8.2 | CS30116 | Computer Networks (Theory Course) | 87 |
| 8.3 | CS30105 | Theory of Computation (Theory Course) | 90 |
| 8.4 | CS30117 | Human Computer Interaction (Theory Course) | 93 |
| 8.5 | CS30109 | Web Technologies (Theory Course) | 96 |
| 8.6 | CS30101 | Operating Systems (Tutorial) | 99 |
| 8.7 | CS30117 | Human Computer Interaction (Tutorial) | 102 |
| 8.8 | CS30303 | Operating Systems (Laboratory Course) | 104 |
| 8.9 | CS30316 | Computer Networks (Laboratory Course) | 105 |
| 8.10 | CS30309 | Web Technologies (Laboratory Course) | 107 |
### Course Structure - Module VI

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Title</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.11 CS37401</td>
<td>Mini Project</td>
<td>110</td>
</tr>
<tr>
<td>8.12 CS30401</td>
<td>Comprehensive Viva Voce</td>
<td>-</td>
</tr>
<tr>
<td>8.14 CS37301</td>
<td>Seminar</td>
<td>111</td>
</tr>
<tr>
<td>8.15 CS37302</td>
<td>Project Stage - I</td>
<td>112</td>
</tr>
</tbody>
</table>

### Course Syllabi for Courses - Module VI

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Title</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.1 CS30102</td>
<td>Software Engineering (Theory Course)</td>
<td>118</td>
</tr>
<tr>
<td>10.2 CS30106</td>
<td>Database Management Systems (Theory Course)</td>
<td>121</td>
</tr>
<tr>
<td>10.3 CS30118</td>
<td>Advanced Data Structures and Algorithms (Theory Course)</td>
<td>124</td>
</tr>
<tr>
<td>10.4 CS30114</td>
<td>Systems Programming (Theory Course)</td>
<td>127</td>
</tr>
<tr>
<td>10.5 CS31119</td>
<td>Object Oriented Modeling and Design (Theory Course)</td>
<td>130</td>
</tr>
<tr>
<td>10.6 CS30102</td>
<td>Software Engineering (Tutorial)</td>
<td>133</td>
</tr>
<tr>
<td>10.7 CS30118</td>
<td>Advanced Data Structures and Algorithms (Tutorial)</td>
<td>135</td>
</tr>
<tr>
<td>10.8 CS30314</td>
<td>System programming (Laboratory Course)</td>
<td>137</td>
</tr>
<tr>
<td>10.9 CS30306</td>
<td>Database Management Systems (Laboratory Course)</td>
<td>139</td>
</tr>
<tr>
<td>10.10 CS31319</td>
<td>Object Oriented Modeling and Design (Laboratory Course)</td>
<td>141</td>
</tr>
<tr>
<td>10.11 CS37402</td>
<td>Mini Project</td>
<td>143</td>
</tr>
<tr>
<td>10.12 CS30402</td>
<td>Comprehensive Viva Voce</td>
<td>-</td>
</tr>
<tr>
<td>10.13 CS37301</td>
<td>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 CS40115</td>
<td>Data Acquisition Systems (Theory Course)</td>
<td>148</td>
</tr>
<tr>
<td>12.2 CS40114</td>
<td>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>
</tbody>
</table>

---

Page 4 of 251
### Course Syllabi for Courses - Module VIII

#### 14.1 CS40112 Software Testing and Quality Assurance (Theory Course)  
- Mobile Computing  
- Cloud Computing  
- Parallel Computing on GPU  
- Advanced Computer Graphics  
- Enterprise Systems  
- Digital Image Processing  

#### 14.2 CS40114 Information Systems Security (Theory Course)  
- Data Mining  
- Machine Learning  
- Neural Networks  

#### 14.3 *Elective Group III (Theory Course)*
- Mobile Computing  
- Cloud Computing  
- Parallel Computing on GPU  
- Advanced Computer Graphics  
- Enterprise Systems  
- Digital Image Processing  

#### 12.4 *Elective Group II (Theory Course)*
- Randomized and Approximation Algorithms  
- Distributed Computing  
- Information Retrieval  
- Digital Signal Processing  

#### 12.5 CS40315 Data Acquisition Systems (Laboratory Course)  
- Mobile Computing  
- Cloud Computing  
- Parallel Computing on GPU  
- Advanced Computer Graphics  
- Enterprise Systems  
- Digital Image Processing  

#### 12.6 *Elective Group II (Tutorial Course)*
- Randomized and Approximation Algorithms  
- Distributed Computing  
- Information Retrieval  
- Digital Signal Processing  

#### 12.7 CS40314 Business Intelligence and Analytics (Tutorial Course)  
- Mobile Computing  
- Cloud Computing  
- Parallel Computing on GPU  
- Advanced Computer Graphics  
- Enterprise Systems  
- Digital Image Processing  

#### 12.8 *Elective Group I (Laboratory Course)*
- Mobile Computing  
- Cloud Computing  
- Parallel Computing on GPU  
- Advanced Computer Graphics  
- Enterprise Systems  
- Digital Image Processing  

#### 12.9 CS47303 Project Stage - II  
- Mobile Computing  
- Cloud Computing  
- Parallel Computing on GPU  
- Advanced Computer Graphics  
- Enterprise Systems  
- Digital Image Processing  

#### Course Structure - Module VIII  

---

Page 5 of 251
<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Name</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS42124</td>
<td>Algorithmic Number Theory and Algebra</td>
<td>220</td>
</tr>
<tr>
<td>CS42118</td>
<td>Geographical Information Systems</td>
<td>224</td>
</tr>
<tr>
<td>CS42129</td>
<td>Management Information Systems</td>
<td>227</td>
</tr>
<tr>
<td>CS42121</td>
<td>Advanced Computer Architecture</td>
<td>230</td>
</tr>
<tr>
<td>CS42116</td>
<td>Convergence Technologies</td>
<td>233</td>
</tr>
<tr>
<td>CS42130</td>
<td>Embedded Systems</td>
<td>235</td>
</tr>
<tr>
<td>CS42116</td>
<td>Geographical Information Systems</td>
<td>224</td>
</tr>
<tr>
<td>CS42129</td>
<td>Management Information Systems</td>
<td>227</td>
</tr>
<tr>
<td>CS42121</td>
<td>Advanced Computer Architecture</td>
<td>230</td>
</tr>
<tr>
<td>CS42116</td>
<td>Convergence Technologies</td>
<td>233</td>
</tr>
<tr>
<td>CS42130</td>
<td>Embedded Systems</td>
<td>235</td>
</tr>
<tr>
<td>CS42124</td>
<td>Algorithmic Number Theory and Algebra</td>
<td>220</td>
</tr>
<tr>
<td>CS42118</td>
<td>Geographical Information Systems</td>
<td>224</td>
</tr>
<tr>
<td>CS42129</td>
<td>Management Information Systems</td>
<td>227</td>
</tr>
<tr>
<td>CS42121</td>
<td>Advanced Computer Architecture</td>
<td>230</td>
</tr>
<tr>
<td>CS42116</td>
<td>Convergence Technologies</td>
<td>233</td>
</tr>
<tr>
<td>CS42130</td>
<td>Embedded Systems</td>
<td>235</td>
</tr>
<tr>
<td>CS42120</td>
<td>Data Mining</td>
<td>237</td>
</tr>
<tr>
<td>CS42128</td>
<td>Machine Learning</td>
<td>238</td>
</tr>
<tr>
<td>CS42104</td>
<td>Neural Networks</td>
<td>240</td>
</tr>
<tr>
<td>CS42124</td>
<td>Algorithmic Number Theory and Algebra</td>
<td>242</td>
</tr>
<tr>
<td>CS42118</td>
<td>Geographical Information Systems</td>
<td>224</td>
</tr>
<tr>
<td>CS42129</td>
<td>Management Information Systems</td>
<td>227</td>
</tr>
<tr>
<td>CS42121</td>
<td>Advanced Computer Architecture</td>
<td>230</td>
</tr>
<tr>
<td>CS42116</td>
<td>Convergence Technologies</td>
<td>233</td>
</tr>
<tr>
<td>CS42130</td>
<td>Embedded Systems</td>
<td>235</td>
</tr>
<tr>
<td>CS40312</td>
<td>Software Testing and Quality Assurance (Laboratory Course)</td>
<td>251</td>
</tr>
<tr>
<td>CS40314</td>
<td>Information Systems Security (Laboratory Course)</td>
<td>253</td>
</tr>
<tr>
<td>CS47308</td>
<td>Project Stage - III</td>
<td>255</td>
</tr>
</tbody>
</table>

15 Course Syllabi for PD Courses in TY B.Tech (Information Technology)
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>
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><strong>GA: 1</strong> Engineering Knowledge</td>
<td>1. To apply scientific and mathematical principles in order to determine conceptual aspects of real world problems in information engineering.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2. To apply algorithmic principles and information science theory for comprehending technological trade-off.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3. To explore conceptual paradigms with incorporation of programming practices.</td>
</tr>
<tr>
<td>P02</td>
<td><strong>GA: 2</strong> Problem Analysis</td>
<td>4. To recognize and synthesize the context of the problem leading to correct and consistent requirements.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>5. To analyze and formulate problem frames in order to receive decomposition structure of information engineering/technology problem.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>6. To identify resources, infrastructure and technology required to realize solution of real world problem.</td>
</tr>
<tr>
<td>P03</td>
<td><strong>GA: 3</strong> Design/Development of solution</td>
<td>7. To plan and devise design alternatives which leads to conceive optimal solution.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>8. To compose technical design specifications for formally expressing the solution implementation.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>9. To practice template based approaches for formulating engineering artifacts addressing information engineering/technology problem.</td>
</tr>
<tr>
<td>P04</td>
<td><strong>GA: 4</strong> Conduct Investigation of Complex Problem</td>
<td>10. To apply research knowledge in order to recognize information engineering/technology problem issues.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>11. To investigate real world information engineering/technology problem with cause effect analysis and inference.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>12. To apply research methods to determine impact and severity of</td>
</tr>
</tbody>
</table>
| PO5          | GA: 5 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          | GA: 6 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          | GA: 7 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          | GA: 8 Ethics | 20. To adhere to ethical responsibility.  
|             |                         | 21. To follow norms of engineering practice.               |
| PO9          | GA: 9 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         | GA: 10 Communication | 24. To demonstrate proficiency in technical and social communications.  
|             |                         | 25. To interpret and represent engineering artifacts considering IT environment.               |
| PO11         | GA: 11 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         | GA: 12 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.
<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>ESA</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Test 1 Test 2 HA Tut. CA ESE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HS10109</td>
<td>Linear Algebra and Random Variables</td>
<td>Theory - Core</td>
<td>3 - 1</td>
<td>10 20 5 5 -</td>
<td>60 4</td>
</tr>
<tr>
<td>HS10103</td>
<td>Modern Physics</td>
<td>Theory – Core</td>
<td>3 - 1</td>
<td>10 20 5 5 -</td>
<td>60 4</td>
</tr>
<tr>
<td>CH10101</td>
<td>Chemistry</td>
<td>Theory – Core</td>
<td>3 - -</td>
<td>15 20 5 - -</td>
<td>60 3</td>
</tr>
<tr>
<td>ME10101</td>
<td>Engineering Graphics</td>
<td>Theory – Core</td>
<td>3 - -</td>
<td>15 20 5 - -</td>
<td>60 3</td>
</tr>
<tr>
<td>HS16101</td>
<td>Sociology</td>
<td>Theory – OE</td>
<td>2 - -</td>
<td>15 20 5 - -</td>
<td>60 2</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>- 2 -</td>
<td>- - - -</td>
<td>70 30 1</td>
</tr>
<tr>
<td>CH10301</td>
<td>Science Lab</td>
<td>Lab – Core</td>
<td>- 2 -</td>
<td>- - - -</td>
<td>70 30 1</td>
</tr>
<tr>
<td>HS17401</td>
<td>Mini Project</td>
<td>Project</td>
<td>- 4 -</td>
<td>- - - -</td>
<td>70 30 2</td>
</tr>
<tr>
<td>TOTAL</td>
<td></td>
<td></td>
<td>14 08 2</td>
<td></td>
<td>20</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</td>
<td>ESA</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Test 1 Test 2 HA Tut. CA ESE</td>
<td></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 - 60</td>
<td>4</td>
</tr>
<tr>
<td>HS10104</td>
<td>Engineering Mechanics</td>
<td>Theory – Core</td>
<td>3   - 1</td>
<td>10  20 5 5 - 60</td>
<td>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 Technology</td>
<td>Theory – OE</td>
<td>2   - -</td>
<td>15  20 5 - -</td>
<td>60 2</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>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></td>
<td><strong>20</strong></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</td>
<td>P</td>
<td>Tut.</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><strong>TOTAL</strong></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   P   Tut.</td>
<td>ISA</td>
<td>ESA</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Test 1 Test 2</td>
<td>HA   Tut.   CA</td>
<td>ESE</td>
</tr>
<tr>
<td>HS17301</td>
<td>General seminar I Comm. Skill</td>
<td></td>
<td>-   2   -   -</td>
<td>-   -   -</td>
<td>70 30 1</td>
</tr>
<tr>
<td>HS153xx</td>
<td>General Proficiency GP</td>
<td></td>
<td>-   2   -   -</td>
<td>-   -   -</td>
<td>70 30 1</td>
</tr>
<tr>
<td>HS15306</td>
<td>French II Language</td>
<td>Workshop</td>
<td>-   2   -   -</td>
<td>-   -   -</td>
<td>70 30 1</td>
</tr>
<tr>
<td>HS15307</td>
<td>German II</td>
<td>Workshop</td>
<td>-   2   -   -</td>
<td>-   -   -</td>
<td>70 30 1</td>
</tr>
<tr>
<td>HS15308</td>
<td>Spanish II</td>
<td>Workshop</td>
<td>-   2   -   -</td>
<td>-   -   -</td>
<td>70 30 1</td>
</tr>
<tr>
<td>HS15309</td>
<td>Japanese II</td>
<td>Workshop</td>
<td>-   2   -   -</td>
<td>-   -   -</td>
<td>70 30 1</td>
</tr>
<tr>
<td>HS15310</td>
<td>English II</td>
<td>Workshop</td>
<td>-   2   -   -</td>
<td>-   -   -</td>
<td>70 30 1</td>
</tr>
<tr>
<td>HS14302</td>
<td>Trade Workshop</td>
<td>Workshop</td>
<td>-   2   -   -</td>
<td>-   -   -</td>
<td>70 30 1</td>
</tr>
<tr>
<td>TOTAL</td>
<td></td>
<td></td>
<td>-   8   -   -</td>
<td>-   -   -</td>
<td>4</td>
</tr>
</tbody>
</table>

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.
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 VI
# Course Structure

## 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>CS30102</td>
<td>Software Engineering</td>
<td>3</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>CS30106</td>
<td>Database Management Systems</td>
<td>3</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>CS30118</td>
<td>Advanced Data Structures and Algorithms</td>
<td>3</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>CS31119</td>
<td>Object Oriented Modeling and Design</td>
<td>3</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>CS30114</td>
<td>Systems Programming</td>
<td>2</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>TOTAL</strong></td>
<td>14</td>
<td>2</td>
<td></td>
</tr>
</tbody>
</table>

## 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>CS30314</td>
<td>System programming</td>
<td>2</td>
<td>30</td>
<td>70</td>
</tr>
<tr>
<td>CS30306</td>
<td>Database Management Systems</td>
<td>2</td>
<td>30</td>
<td>70</td>
</tr>
<tr>
<td>CS31319</td>
<td>Object Oriented Modeling and Design</td>
<td>2</td>
<td>30</td>
<td>70</td>
</tr>
<tr>
<td>CS37402</td>
<td>Mini Project (T.Y.Semester I)</td>
<td>2</td>
<td>30</td>
<td>70</td>
</tr>
<tr>
<td></td>
<td>Irrespective of Module</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CS37301</td>
<td>Seminar(T.Y. Semester I)</td>
<td>4</td>
<td>30</td>
<td>70</td>
</tr>
<tr>
<td></td>
<td>Irrespective of Module</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PD</td>
<td>Institute Level Elective</td>
<td>Credits</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-----</td>
<td>--------------------------------------------------</td>
<td>---------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CS30402</td>
<td>Comprehensive Viva Voce Based on Courses S2, S4</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CS37302</td>
<td>PROJECT STAGE 1 (T.Y. Semester II) Irrespective of Module</td>
<td>4</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

| TOTAL | | 16 | 10 |

<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 Laboratory</td>
</tr>
<tr>
<td>^</td>
<td>Detail Enclosed Separately</td>
</tr>
</tbody>
</table>
CS 30102: Software Engineering

Credits: 03

Teaching Scheme: - Theory 3 Hrs/Week

Prerequisites: Data Structures

Unit 1: Software Engineering Paradigms (8 Hrs)

Unit 2: Requirement Engineering (8 Hrs)
Part B: Requirements Verification and Validation, Requirement Maturity, Technical Reviews

Unit 3: System Analysis and Design Foundations (8 Hrs)

Unit 4: System Architecture Determination (8 Hrs)

Part B: Architecture Analysis Techniques, Zachman Framework, Architecture Assessment

Unit 5: Project Management Principles (8 Hrs)


Part B: Classic Mistakes, Complex Systems, Critical Systems, Software Safety

Text Books:

Reference Books:

Course Outcomes:

Upon completion of the course, graduates will be able to –
1. Identify the process models required to construct software in order to quench stakeholder needs and requirements.
2. Interpret the problem scope associated with real world problems.
3. Compose software artifacts with conformation to stated requirements.
4. Evaluate the criteria required to balance overall problem solution pair.
5. Demonstrate synergistic and cohesive team work that justifies solution realizations.
6. Build realistic solution assembled by either technological availability or through creativity.
CS30106:: DATABASE MANAGEMENT SYSTEMS

Credits: 03

Teaching Scheme: - Theory 3 Hrs/Week

Prerequisites: Data structures

Unit 1: Introduction and Data Models (9+2 Hrs)

Data Models: Entity Relationship (ER) Model, Extended ER Model, Relational Data Model, Object Oriented Data model, Semi-structured Data Model: DTD or XML Schema
PART B: Spreadsheet Model, Codd's Twelve Rules for Relational DBMS, Life Cycle of a Relational Database

Unit 2: Database Design Theory (7+2 Hrs)

PART A: Normalization: Need, Functional Dependency, Inference Rules, FD Closure, Minimal Cover, Decomposition Properties, Normal Forms (upto BCNF), Multi-valued Dependency (4NF), Relational Synthesis Algorithm
PART B: Join and Inclusion Dependency, 5NF, DKNF, Trade-off

Unit 3: Query Languages (6+2 Hrs)

PART A: Formal Relational Query Languages: Relational Algebra, Tuple Relational Calculus;
SQL: DDL, DML, Select Queries, Join Queries, Subqueries;
PL/SQL: Procedure, Function, Trigger;
Query by Example;
PART B: Domain Relational Calculus, DCL-Security and Authorization, Date-Timestamp, String and Numerical Functions, Mapping of Relational Algebra to SQL

Unit 4: Storage and Querying (9+2 Hrs)

PART A: Storage: Storage and File structure, Files with Fixed / Variable Length Records, Hashed Files;
Indexing: Indexed Files, Single Level and Multi Level Indexes, B+ Trees;
Query Processing: Steps, Algorithms for Selection, Join Operation;
Query Optimization: Transformation of Relational Expressions, Choice of Evaluation Plans; Query Execution Cost;
PART B: SAN, Files with Sparse / Dense Index; Query Processing: Sort Operation, Impact of Indices on Query Performance;

Unit 5: Transaction Management and Emerging Trends (9+2 Hrs)

Page 26 of 251
PART A: Transaction: ACID Properties, Concurrency Control Protocols: Lock-based, Multiple Granularity, Multiversion Scheme; Failure and Recovery; NoSQL: RDBMS vs NoSQL, BASE properties, NoSQL Categories; NewSQL; Emerging Trends: Distributed Databases, Distributed Data Storage, Distributed Query Processing; Parallel Databases, Architectures, Speedup and Scaleup, Decomposition, Data Replication; Time Series Databases, Spatial and Geographic Databases;

PART B: Design of Core DBMS Functions, Timestamp based Concurrency Control Protocol, ARIES Recovery Technique, Personal Databases

Text Books

Reference Books
3. "Getting Started with NoSQL: Your guide to the world and technology of NoSQL", by Gaurav Vaish

Course Outcomes:
Upon completion of the course, graduates will be able to -
1. Develop a database system using relational database query languages, PL/SQL and NoSQL.
2. Construct refined logical database model with consideration of data semantics and dependency.
3. Design data models to enforce data requirements and operational constraints of an organization.
4. Describe techniques used by a DBMS for data storage, access and query processing.
5. Describe various database system architectures and their functionalities.
6. Formulate alternative queries for given data requirement considering the query evaluation plan.
CS30118 :: ADVANCED DATA STRUCTURES AND ALGORITHMS

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

Prerequisites: Data Structures

Unit 1: Advanced searching and retrieval data structures (8+1 Hrs)
PART B: Alternate hash functions (mid-square, folding, digit analysis), LCS problem

Unit 2: Priority Queues and Advanced Heap structures (8+1 Hrs)
PART B: Comparative study of different priority queue implementations using – binary heaps, leftist trees, binomial heaps, Fibonacci heaps with respect to the following operations – insert, delete, find-min, extract-min, decrease-key, meld

Unit 3: Overview of Time Complexity analysis, Divide and Conquer (8+1 Hrs)
PART A: Asymptotic notations (Big Oh, small oh, Big Omega, Theta notations) and time complexity, Overview of searching, sorting algorithms (binary search, insertion sort, heap sort, bubble sort), lower bound for comparison based sorting. Using Recurrence relations and Mathematical Induction to get asymptotic bounds on time complexity and to prove correctness of algorithms.
PART B: Finding Maximum and Minimum, Convex Hull problem, Master’s Theorem’s and its cases

Unit 4: Greedy Strategy and Dynamic Programming (8+1 Hrs)
PART B: String Editing Problem, Scheduling problem, Optimal Storage Problem.

Unit 5: NP-Theory (8+1 Hrs)


PART B: Decision Vs Search versions of problems in class NP, Coupon Collector Problem, Bin Packing.

Text Books

Reference Books

Course Outcomes:

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

1. Interpret the design and implementation of advanced data structures.
2. Analyze the time and space complexity algorithms using asymptotic notations.
3. Apply appropriate algorithm design technique and data structures to solve problems.
4. Grasp the notion of intractability, NP- Complete and NP-hard problems.
5. Appraise the role of randomization and approximation in computation.
6. Make intelligent decisions about data structures and algorithmic techniques in the context of practical software problems, choosing from existing data structures and algorithms or designing their own when necessary.
CS30114:: SYSTEMS PROGRAMMING

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

Prerequisites:
- Data Structures
- Computer Organization
- Microprocessors

Unit 1: Introduction to System Programming (6+1 Hrs)
Assemblers: Elements of Assembly language programming, Simple assembler scheme, Structure of an assembler, Design of single and two pass assembler.
Macro Processors: Macro Definition and call, Macro expansion, Nested Macro Calls, Advanced Macro Facilities, design of Macro Preprocessor. Booting Procedure for DOS & Windows.
PART B: RISC machines, Machine dependent and machine independent Assembler features.

Unit 2: Compilers, Loaders and Linkers (5+1 Hrs)
PART A: Compilers: Introduction to Compiler phases, Introduction to cross compiler, Features of machine dependent and independent compilers, Overview of types of compilers, Interpreters.
Linkers: Relocation and linking concepts, Static and dynamic linker, subroutine linkages.
Loaders: Introduction to Loader, Loader Schemes: Compile and go, General Loader Scheme, Absolute loaders, relocating loaders, direct linking loaders.
PART B: Instruction description, Pseudo operations, Instruction Mapping, MSDOS Linker, Sun OS linker.

Unit 3: Essential concepts of Systems programming for Linux as Open Source OS.
PART A: Introduction and essential concepts of LINUX system programming: System Programming, APIs and ABIs, standards, Program segments/sections; The ELF Format, Linking and loading, Linux dynamic libraries (shared objects), Multitasking and paging, Address translation, Memory Protection, Comparison with Windows.

PART B: Dynamic linking, API compatibility, Dynamically linked libraries, Overall architecture and limitations.

Unit 4: Encoding, Decoding and Device drivers


PART B: Library Description for IA-32/Intel64.

Unit 5: TSR Programming


Text Books

Reference Books

Course Outcomes

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

1. Develop different system software like Macroprocessor, Assembler, Compiler, Linker and Loader.
2. Discriminate among different System software and their functionalities.
3. Design Device Drivers, TSR programs and DLL for real world applications.
4. Solve critical problems related to Encoding, Decoding and Instruction set.
5. Conforms to use proper data structures for system programming.
6. Follow methods and techniques for implementing system-level programs.
CS31119 : Object Oriented Modeling and Design

Credits: 03

Teaching Scheme: - Theory 3 Hrs/Week

Prerequisites: Data Structures

Unit I : Introduction to Modeling (8 Hrs)
Part B: Methodology: Coad-Yordon, Responsibility-Driven Design, OPM, Catalysis, Aspect-Oriented Modeling

Unit 2 : Model Driven Development (8 Hrs)
Part B: Domain-Specific Modeling: Fundamentals and Architecture, MDA Applications

Unit 3 : Behavior Specification (8 Hrs)
Part A: Static Behavior: Use Cases, Use Case Diagram Components, Use Case Diagram, Actor Generalization, Include and Extend, Template for Use Case Narrative, Using Use Cases, The Domain Perspective, Data Dictionary: Finding the Objects, Responsibilities, Collaborators, and Attributes, CRC Cards, Class Models and Use Case Models, Judging the Domain Model, Capturing system behavior in use cases
Dynamic Behavior: Sequence diagrams, object lifelines and message types, Modeling collections multiobjects, Refining sequence diagrams, Collaboration diagrams, States, events and actions, Nested machines and concurrency, Modifying the object model to facilitate states, Modeling methods with activity diagrams, Activity Diagrams: Decisions and Merges, Synchronization, Iteration, Partitions, Parameters and Pins, Expansion Regions, Swimlanes, concurrency and synchronization
Part B: Study of other Behavioral Diagrams: Communication Diagram, Interaction Overview Diagrams, Timing Diagrams

Unit 4 : Design Specification (8 Hrs)
Part A: Design of Software Objects, Features and Methods, Cohesion and Coupling between Objects, Coupling and Visibility, Interfaces, Interfaces with Ball and Socket Notation, Templates, Analysis model vs. design model classes, Categorizing classes:
entity, boundary and control , Modeling associations and collections, Preserving referential integrity , Achieving reusability, Reuse through delegation, Identifying and using service packages, Improving reuse with design Packages and interfaces: Distinguishing between classes/interfaces, Exposing class and package interfaces, Subscribing to interfaces Component and deployment diagrams: Describing dependencies, Deploying components across threads, processes and processors
\Forward Engineering and Reverse Engineering Concepts
Part B: Application of UML in Real Time and Embedded System, Application of UML in Web Engineering, UML Profiles for other technology disciplines

**Unit 5 : Design Patterns**


Part B: Antipatterns, Applications of Design Patterns, Archetype Patterns

**Text Books:**


**Reference Books:**


**Course Outcomes:**

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

1. Determine the unfamiliarity of the problem frames in order to envisage conceptual nomenclature.
2. Break down system functionalities into realizable customer-centric and developer-centric situations.
3. Narrate design specifications in terms of industries-specific practices such as methodology –driven engineering.
4. Propose multi-faceted defendable solutions with overt-behavior demonstrating team-skills.
5. Initiate new problem issues and compatible solution aspects with the help of design pattern.
6. Automatically devise solution terminologies reducing the potential of cost and performance impedance.
CS 30102: Software Engineering

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

Prerequisites: Data Structures

List of Contents

A TERM-WORK containing the record of the following:

1. To study Software Process Models and identify their applicability to various categories of projects.

2. To understand Requirement Elicitation Techniques and recognize types of requirements while preparing System Requirement Specification.


4. To develop all level Data Flow diagrams for the target system indicating problem partitions and solution structure.

5. To apply design principles with relevant architecture style and structure the solution accordingly.

6. To prepare estimation for the System Development using Function Point technique.

Text Books:

Reference Books :
CS30118:: Advanced Data Structures And Algorithms

Credits: 01

Teaching Scheme: - Tutorial 1 Hr/Week

Prerequisites: : Data Structures

List of Contents

1. Example to indicate the limitations of static hashing and how it is addressed using dynamic hashing.
2. Implement an application that uses bloom filter.
3. Improve the performance of Dijkstra’s shortest path algorithm using Fibonacci heaps.
4. Implement a binomial heap and compare its amortized complexity with binary heap.
5. Implement an application that makes use of a prefix tree (trie) – address book, spell checker, auto completion etc.
6. Implement a solution for LCS problem (Longest Common Subsequence) using suffix trees.
7. Numerical Problems based on
   a. Mathematical Induction,
   b. Solving recurrence relations,
   c. Proof by contradiction.
8. Time complexity comparison of matrix multiplication by using
   b. Strassens Matrix multiplication
10. Implementation of travelling sales person problem using Dynamic programming technique.
11. Implementation of OBST or 0/1 knapsack problem using dynamic programming technique.
12. Implementation and time complexity comparison of Randomized and non-randomized version of algorithm for MIN_CUT problem by using different test cases.
13. Implementation and time complexity comparison of Randomized and non-randomized version of algorithm for Quick sort problem by using different test cases.
15. Introduction to Linear programming and study of (Simplex Algorithm, Ellipsoid Algorithm, Interior Point Algorithm).
16. Study of parallel algorithms and parallel programming techniques.

Text Books


CS30314:: SYSTEMS PROGRAMMING

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

List of Assignments:

1. Expanding the Simple Macros with Generating different Parameter Tables and MDT.
2. Expanding the Nested Macros with Generating different Parameter Tables and MDT.
3. Design and implementation of 1 pass and 2 Pass assemblers with generating different data structure for it.
4. Design and implementation of an Editor: Design of a Line or Screen Editor using C Language.
5. Symbol table generation for input *.c file.
7. Simulation of linkers.
8. Simulation of loaders.
9. Understanding the design for DLL on Linux shared library.
10. Use of different debugger tools.
11. Printer controller in device drivers.
12. Write a TSR program in 8086 ALP to implement Real Time Clock (RTC). Read the Real Time from CMOS chip by suitable INT and FUNCTION and display the RTC at the bottom right corner on the screen. Access the video RAM directly in your routine.
13. Write a TSR program in 8086 ALP to implement Screen Saver. Screen Saver should get activated if the keyboard is idle for 7 seconds. Access the video RAM directly in your routine.
14. Write a TSR program in 8086 ALP to handle the “Divide by zero” interrupt. Test your program with a small code, which causes the divide by zero interrupt.
15. Write a TSR program in ‘C’ that would change the color of the screen every 10 seconds.

Note: It is expected that student must perform at least 2 assignments from assignment number 12 to 15.

Text Books

Reference Books
CS30306:: DATABASE MANAGEMENT SYSTEMS

Credits: 01

Teaching Scheme: - Laboratory 2 Hrs/Week

Prerequisites:

List of Practical

1. Choose a database system you propose to work on throughout the course. Perform requirements analysis in detail for design of the database. Design an entity-relationship (ER) data model for the selected database system.

2. Convert above ER model to relational model, semi_structured data model. List functional dependencies. Normalize these relations up to 3NF/BCNF.

3. Consider a different database system. List functional dependencies [Include complex business logic.] Apply bottom_up approach using Relational Synthesis Algorithm for design of relational model for the chosen system. Verify decomposition properties.

4. Create tables with appropriate constraints for the relational schema. Create views, indices, and sequence. Alter the schema by adding/removing columns and constraints. Write DML queries.

5. Execute ‘SELECT’ queries using order by, group by, aggregate functions, having clause, and set operators. Use SQL single row functions for date, time, string etc.

6. Write equijoin, non equijoin, self join and outer join queries. Write queries containing single row / multiple row / corelated subqueries using operators like =, in, any, all, exists etc. Write DML queries containing subqueries. Study a set of query processing strategies.

7. Write meaningful stored procedures in PL/SQL. Make use of cursors and different arguments. Write useful stored functions to perform complex computation. Write row level and statement level triggers in PL/SQL.

8. Implement a small database application for the above system using suitable front end and back end tool. Create a transaction by embedding SQL into an application program. Generate different useful reports.

9. Implementation of a small database using NoSQL and/or New SQL database system.
Text Books


Reference Books


3. "Getting Started with NoSQL: Your guide to the world and technology of NoSQL", by Gaurav Vaish

CS31319: Object Oriented Modeling and Design

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

Prerequisites: Data Structures

List of Practical

1. To narrate Requirement Definition Document for the target system with following three areas:
   a. Problem Identification
   b. Problem Definition
   c. Problem Statement

2. To decompose and organize the problem domain area into broad subject areas and identify the boundaries of problem/system. Specify the behavior of the target system and map requirements to Use cases. The System Context Diagram depicts the overall System behavioral trace and Requirement Capture diagram depicts the hierarchical Use case Organization. The Use Case diagram should encompass:
   a. Actors (External Users)
   b. Transactions (Use Cases)
   c. Event responses related to transactions with external agents.
   d. Detection of System boundaries indicating scope of system.

3. To depict the dynamic behavior of the target system using sequence diagram. The Sequence diagram should be based on the Scenarios generated by the inter-object Communication. The model should depict:
   a. Discrete, distinguishable entities (class).
   b. Events (Individual stimulus from one object to another).
   c. Conditional events and relationship representation.

4. To depict the state transition with the life history of objects of a given class model. The model should depict:
   a. Possible ways the object can respond to events from other objects.
   b. Determine of start, end, and transition states.

5. To depict the dynamic behavior using detailed Activity diagram.

6. To develop logical static structure of target system with Software Class diagram. To prepare Class Collaboration-Responsibility (CRC) cards for the Conceptual classes traced from System analysis phase. The design model should depict...
a. Relationship between classes: inheritance, Assertion, Aggregation, Instantiation
b. Identification of objects and their purpose.
c. Roles / responsibilities entities that determine system behavior.

7. To represent physical module that provides occurrence of classes or other logical elements identified during analysis and design of system using Component diagram. The model should depict allocation of classes to modules. To narrate precise Program Design Language constructs separating computation from interface. To represent deployment view of the system through Architecture Diagram.

8. To enhance Software Class diagram to Architecture diagram with appropriate design patterns. To implement the system according to specification with confirmation to design patterns.

Text Books:

Reference Books:
CS37402::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.
Students are expected to choose a topic in CSE based on current trends or industry practices. They should prepare present a report and present it after having fully understood the concepts. Evaluation will be based on relevance of topic, 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.
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](http://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 NetBeans 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.

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 Strus Live" by Rick Hightower published by SourceBeat.

Reference Books


Course Outcomes:

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

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


Page 62 of 251
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.