MVN UNIVERSITY

Department of Computer Science and Engineering

Integrated Master of Technology

Scheme And Syllabus
### Semester-I

<table>
<thead>
<tr>
<th>S. No</th>
<th>Course Title</th>
<th>Paper Code</th>
<th>Teaching Schedule</th>
<th>Total</th>
<th>Credit</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Applied Mathematics – I</td>
<td>AHL 103</td>
<td>3 1 0</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>Applied Physics – I</td>
<td>AHL 101</td>
<td>3 1 2</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>Communication Skills in English - I</td>
<td>AHL 105</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>Introduction to Computer Programming</td>
<td>CSL101</td>
<td>3 1 2</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td>Electrical Technology / Basic Electronics</td>
<td>EEL100/ECL100</td>
<td>3 1 2</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>6</td>
<td>Elements of Mechanical Engineering</td>
<td>MEL 101</td>
<td>3 1 2</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>7</td>
<td>Professional Communication-I</td>
<td>AHP101</td>
<td>1 0 2</td>
<td>3</td>
<td>NC</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td></td>
<td>19 5 10</td>
<td>34</td>
<td>27</td>
</tr>
</tbody>
</table>

### Semester-II

<table>
<thead>
<tr>
<th>S. No</th>
<th>Course Title</th>
<th>Paper Code</th>
<th>Teaching Schedule</th>
<th>Total</th>
<th>Credit</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Applied Mathematics – II</td>
<td>AHL 104</td>
<td>3 1 0</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>Applied Physics – II</td>
<td>AHL 102</td>
<td>3 1 2</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>Communication skills in English - II</td>
<td>AHL 106</td>
<td>3 0 0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>Advanced C Programming</td>
<td>CSL 102</td>
<td>3 1 2</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td>Engineering Drawing</td>
<td>MEP102</td>
<td>1 0 3</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>6</td>
<td>Electrical Technology/Basic Electronics</td>
<td>EEL 100/ECL 100</td>
<td>3 1 2</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>7</td>
<td>Professional Communication-II</td>
<td>AHP102</td>
<td>1 0 2</td>
<td>3</td>
<td>NC</td>
</tr>
<tr>
<td>8</td>
<td>Environmental Studies</td>
<td>AHL108</td>
<td>2 0 0</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td></td>
<td>19 4 11</td>
<td>34</td>
<td>27</td>
</tr>
</tbody>
</table>
## Semester-I

<table>
<thead>
<tr>
<th>S. No</th>
<th>Course Title</th>
<th>Paper Code</th>
<th>Teaching Schedule</th>
<th>Total</th>
<th>Credit</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Object Oriented Programming using C++</td>
<td>CSL203</td>
<td>3 L 1 T 2 P</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>Data Structure &amp; Applications</td>
<td>CSL205</td>
<td>3 L 1 T 2 P</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>Discrete Structures</td>
<td>CSL207</td>
<td>3 L 1 T 0 P</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>Digital Electronics</td>
<td>ECL203</td>
<td>3 L 1 T 2 P</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td>Managerial Economics</td>
<td>MSL503</td>
<td>3 L 0 T 0 P</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>6</td>
<td>Applied Mathematics-III</td>
<td>AHL211</td>
<td>3 L 1 T 0 P</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>7</td>
<td>Professional Communication – III</td>
<td>AHP201</td>
<td>1 L 0 T 2 P</td>
<td>3</td>
<td>NC</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td></td>
<td>19 L 5 T 8 P</td>
<td>32</td>
<td>26</td>
</tr>
</tbody>
</table>

## Semester-IV

<table>
<thead>
<tr>
<th>S. No</th>
<th>Course Title</th>
<th>Paper Code</th>
<th>Teaching Schedule</th>
<th>Total</th>
<th>Credit</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Computer Architecture &amp; Organization</td>
<td>CSL 202</td>
<td>3 L 1 T 0 P</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>Database Admin Concepts &amp; Management</td>
<td>CSL 204</td>
<td>3 L 1 T 2 P</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>Software Engineering Concepts</td>
<td>CSL 206</td>
<td>3 L 0 T 0 P</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>Core Java</td>
<td>CSL 208</td>
<td>3 L 1 T 2 P</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td>Internet &amp;Web Technology</td>
<td>CSL 212</td>
<td>3 L 0 T 2 P</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>Programming Language</td>
<td>CSL 214</td>
<td>3 L 0 T 0 P</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>7</td>
<td>VAC-I</td>
<td>CSV202</td>
<td>0 L 0 T 2 P</td>
<td>2</td>
<td>NC</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td></td>
<td>18 L 3 T 8 P</td>
<td>29</td>
<td>24</td>
</tr>
</tbody>
</table>
### Integrated Master of Technology in Computer Science and Engineering

#### Semester-V

<table>
<thead>
<tr>
<th>S. No</th>
<th>Course Title</th>
<th>Paper Code</th>
<th>Teaching Schedule</th>
<th>Total</th>
<th>Credit</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>L</td>
<td>T</td>
<td>P</td>
</tr>
<tr>
<td>1</td>
<td>Operating System</td>
<td>CSL301</td>
<td>3</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>Multimedia &amp; Animation</td>
<td>CSL303</td>
<td>3</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>FL &amp; Automata Theory</td>
<td>CSL305</td>
<td>3</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>Algorithm Design &amp; Analysis</td>
<td>CSL307</td>
<td>3</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>5</td>
<td>Microprocessor &amp; Interfacing</td>
<td>ECL307</td>
<td>3</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>Computer Graphics</td>
<td>CSL309</td>
<td>3</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>7</td>
<td>Practical Training -I</td>
<td>CST311</td>
<td>0</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>8</td>
<td>Professional Communication – IV</td>
<td>AHP301</td>
<td>1</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>19</td>
<td>5</td>
<td>12</td>
</tr>
</tbody>
</table>

#### Semester-VI

<table>
<thead>
<tr>
<th>S. No</th>
<th>Course Title</th>
<th>Paper Code</th>
<th>Teaching Schedule</th>
<th>Total</th>
<th>Credit</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>L</td>
<td>T</td>
<td>P</td>
</tr>
<tr>
<td>1</td>
<td>Distributed Operating System</td>
<td>CSL302</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>Artificial Intelligence</td>
<td>CSL 304</td>
<td>3</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>Principles of Compiler Design</td>
<td>CSL306</td>
<td>3</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>Computer Networks</td>
<td>CSL308</td>
<td>3</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>5</td>
<td>Introduction to Cyber Security</td>
<td>CSL310</td>
<td>3</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td>Advanced java</td>
<td>CSL312</td>
<td>3</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>7</td>
<td>VAC –II</td>
<td>CSV302</td>
<td>0</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>18</td>
<td>3</td>
<td>12</td>
</tr>
<tr>
<td>S. No</td>
<td>Course Title</td>
<td>Paper Code</td>
<td>Teaching Schedule</td>
<td>Total</td>
<td>Credit</td>
</tr>
<tr>
<td>-------</td>
<td>--------------------------------------------------</td>
<td>------------</td>
<td>-------------------</td>
<td>-------</td>
<td>--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>L</td>
<td>T</td>
<td>P</td>
</tr>
<tr>
<td>1</td>
<td>Network Programming</td>
<td>CSL403</td>
<td>3</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>Neural Networks</td>
<td>CSL405</td>
<td>3</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>Software Engineering Concepts &amp; Methodology</td>
<td>CSL503</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>Concepts of Distributed Systems</td>
<td>CSL505</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>Advanced C/C++ Programming</td>
<td>CSL507</td>
<td>3</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td>Mat Lab</td>
<td>CSP509</td>
<td>0</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>7</td>
<td>Seminar-I</td>
<td>CSS401</td>
<td>0</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>8</td>
<td>Minor Project</td>
<td>CSD401</td>
<td>0</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>9</td>
<td>Practical Training-II</td>
<td>CST411</td>
<td>0</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>10</td>
<td>Professional Communication-V</td>
<td>AHP401</td>
<td>1</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td></td>
<td>16</td>
<td>0</td>
<td>18</td>
</tr>
</tbody>
</table>
## Semester-VIII

<table>
<thead>
<tr>
<th>S. No</th>
<th>Course Title</th>
<th>Paper Code</th>
<th>Teaching Schedule</th>
<th>Total</th>
<th>Credit</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>L</td>
<td>T</td>
<td>P</td>
</tr>
<tr>
<td>1</td>
<td>Advanced Computer Architecture</td>
<td>CSL402</td>
<td>3</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>Cryptography and Network Security</td>
<td>CSL404</td>
<td>3</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>Software Project Management</td>
<td>CSL406</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>Advanced Operating System</td>
<td>CSL504</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>Elective-I</td>
<td>1*</td>
<td>3</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td>Oracle Lab</td>
<td>CSP512</td>
<td>0</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>7</td>
<td>Seminar-II</td>
<td>CSS609</td>
<td>0</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>8</td>
<td>Project</td>
<td>CSD408</td>
<td>0</td>
<td>0</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td><strong>Total</strong></td>
<td></td>
<td>15</td>
<td>2</td>
<td>14</td>
</tr>
</tbody>
</table>

### Elective I

<table>
<thead>
<tr>
<th>S. No</th>
<th>Course Title</th>
<th>Paper Code</th>
<th>Teaching Schedule</th>
<th>Total</th>
<th>Credit</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>L</td>
<td>T</td>
<td>P</td>
</tr>
<tr>
<td>1</td>
<td>Concepts of Soft Computing</td>
<td>CSL516</td>
<td>3</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>Concepts of Information Retrieval</td>
<td>CSL518</td>
<td>3</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>Introduction to Object oriented Software Engineering</td>
<td>CSL526</td>
<td>3</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>Concepts of Data Warehousing and Data Mining</td>
<td>CSL524</td>
<td>3</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td><strong>Total</strong></td>
<td></td>
<td>3</td>
<td>0</td>
<td>2</td>
</tr>
</tbody>
</table>

## Semester-I X

<table>
<thead>
<tr>
<th>S. No</th>
<th>Name of subject</th>
<th>Sub code</th>
<th>Teaching Schedule</th>
<th>Total</th>
<th>Credit</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>L</td>
<td>T</td>
<td>P</td>
</tr>
<tr>
<td>1</td>
<td>Advanced JAVA</td>
<td>CSL506</td>
<td>3</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>Grammar and Natural Language Processing</td>
<td>CSL508</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>Information System Security</td>
<td>CSL601</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>Algorithm Design and Techniques</td>
<td>CSL603</td>
<td>3</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>5</td>
<td>Elective-II</td>
<td>2*</td>
<td>3</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td>Dissertation-I</td>
<td>CSD601</td>
<td>0</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>7</td>
<td><strong>Total</strong></td>
<td></td>
<td>15</td>
<td>0</td>
<td>10</td>
</tr>
</tbody>
</table>

### (2*) Elective-II

<table>
<thead>
<tr>
<th>S. No</th>
<th>Name of subject</th>
<th>Sub code</th>
<th>Teaching Schedule</th>
<th>Total</th>
<th>Credit</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>L</td>
<td>T</td>
<td>P</td>
</tr>
<tr>
<td>1</td>
<td>Advanced Soft Computing</td>
<td>CSL607</td>
<td>3</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>Advanced Information Retrieval</td>
<td>CSL609</td>
<td>3</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>Advanced Software Engineering</td>
<td>CSL611</td>
<td>3</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>Advanced Data Warehousing and Data Mining</td>
<td>CSL613</td>
<td>3</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td><strong>Total</strong></td>
<td></td>
<td>3</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>S. No</td>
<td>Course Title</td>
<td>Paper Code</td>
<td>Teaching Schedule</td>
<td>Total</td>
<td>Credit</td>
</tr>
<tr>
<td>-------</td>
<td>--------------------</td>
<td>------------</td>
<td>-------------------</td>
<td>-------</td>
<td>--------</td>
</tr>
<tr>
<td>1</td>
<td>Dissertation-II</td>
<td>CSD602</td>
<td>0 0 22</td>
<td>22</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td>Total</td>
<td></td>
<td>0 0 22</td>
<td>22</td>
<td>11</td>
</tr>
<tr>
<td>S. No</td>
<td>Course Title</td>
<td>Paper Code</td>
<td>Teaching Schedule</td>
<td>Total</td>
<td>Credit</td>
</tr>
<tr>
<td>-------</td>
<td>--------------------------------------------</td>
<td>------------</td>
<td>-------------------</td>
<td>-------</td>
<td>--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>L</td>
<td>T</td>
<td>P</td>
</tr>
<tr>
<td>1</td>
<td>Applied Mathematics – I</td>
<td>AHL 103</td>
<td>3</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>Applied Physics – I</td>
<td>AHL 101</td>
<td>3</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>Communication Skills in English - I</td>
<td>AHL 105</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>Introduction to Computer Programming</td>
<td>CSL101</td>
<td>3</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>5</td>
<td>Electrical Technology / Basic Electronics</td>
<td>EEL100/ECL100</td>
<td>3</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td>Elements of Mechanical Engineering</td>
<td>MEL 101</td>
<td>3</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>7</td>
<td>Professional Communication-I</td>
<td>AHP101</td>
<td>1</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Total</td>
<td></td>
<td>19</td>
<td>5</td>
<td>10</td>
</tr>
</tbody>
</table>

**Semester-II**

<table>
<thead>
<tr>
<th>S. No</th>
<th>Course Title</th>
<th>Paper Code</th>
<th>Teaching Schedule</th>
<th>Total</th>
<th>Credit</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>L</td>
<td>T</td>
<td>P</td>
</tr>
<tr>
<td>1</td>
<td>Applied Mathematics – II</td>
<td>AHL 104</td>
<td>3</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>Applied Physics – II</td>
<td>AHL 102</td>
<td>3</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>Communication skills in English - II</td>
<td>AHL 106</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>Advanced C Programming</td>
<td>CSL 102</td>
<td>3</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>5</td>
<td>Engineering Drawing</td>
<td>MEP102</td>
<td>1</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>6</td>
<td>Electrical Technology/Basic Electronics</td>
<td>EEL 100/ECL 100</td>
<td>3</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>7</td>
<td>Professional Communication-II</td>
<td>AHP102</td>
<td>1</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>8</td>
<td>Environmental Studies</td>
<td>AHL108</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>Total</td>
<td></td>
<td>19</td>
<td>4</td>
<td>11</td>
</tr>
</tbody>
</table>
Objective
Today is the era of Computer. This subject focuses on the introduction of Computer to each student of every discipline.

Theory
Note: Total five questions are to be attempted from Sections A & B.

SECTION – A

Unit 1: Introduction
Introduction to Computer, functional units of computer, Types of memories, Introduction to microprocessor, Number system.

Unit 2: Operating system
Introduction to Operating System and its functions, Type of languages like low level, middle level, assembly language and high level, Introduction to Compiler, interpreter, assembler, loader and linker.

Unit 3: Networking
Introduction to Computer Network and various topologies, Introduction to LAN, MAN and WAN.

SECTION – B

Unit 4: Introduction to C
Introduction, Constants, Variables and Data types, Operators and Expressions, Managing I/O operations, Decision Making and branching, Decision Making and looping.

Unit 5: Arrays
Arrays, Character Arrays and Strings, Library and user defined functions. Pointers and its use.

Unit 6: Structure and Union
Defining structure, declaring variables, Accessing structure members, structure initialization, copying and comparing structure variables, operations on individual members, Array of structure, structure with structure, unions and size of structure.
Text Books

2. Computer Fundamental & C programming by J.B.Dixit ; University Science Press
3. Fundamentals of Computer by V Rajaraman; Prentice Hall of India Pvt. Ltd., New Delhi

Reference Books

1. Let Us C by Yaswant Kanetkar : BPB Publication
2. Computer Fundamental & C programming by E. Balaguruswamy; MGH
3. Computers Today by SK Basandara, Galgotia publication Pvt ltd. Daryaganj, New Delhi

LAB:
Note: At least ten experiments are to be performed during the semester. At least eight experiments should be performed from the list of experiments. Two experiments may either be performed from the given list of experiments or may be designed by the concern faculty in consultation with H.O.D as per the scope of syllabus.

Objective: To understand the concepts of programming.

List of Experiments:
Note: C/C++ can be used to implement the following programs.
1- Program to demonstrate the use of variables, and input output statements.
2- Program to demonstrate the use of various arithmetic and logical operators.
3- Program to demonstrate the use of various decision making statements.
4- Program to demonstrate the use of various looping statements.
5- Program to demonstrate the implementation of one dimensional array and its various operations
6- Program to demonstrate the implementation of two dimensional arrays and its various operations
7- Program to demonstrate the implementation of pointers and its arithmetic’s
8- Program to demonstrate the implementation of call by reference and call by value mechanism.
9- Program to demonstrate the implementation of structure.
10- Program to demonstrate the array of structure.
11- Program to demonstrate the use of union.
Objective
This subject focuses better understanding and deeper knowledge of the advanced features of the C programming language.

Theory
Note: Total five questions are to be attempted from Sections A & B.

SECTION – A

Unit 1: String
String introduction, String declaration, Reading and writing strings, String manipulation functions: concatenation, copy, converting in upper case and vice versa, reversing string, comparing string, finding length of string.

Unit 2: Pointer introduction
Introduction, Understanding Pointers, Accessing the address of a variable, Declaring Pointer Variables, Initialization of Pointer Variables, Accessing a variable through its pointer, Chain of Pointers.

Unit 3: Pointers with arrays and function
Pointer Expressions, Pointer Increments and Scale Factors, pointers and Arrays, Pointer and Character Strings, Arrays of Pointers, Pointers as Function Arguments, Functions Returning Pointers, Pointers to structure.

SECTION – B

Unit 4: Memory Management
Introduction, storage classes, Dynamic memory allocation, allocating a block of memory: Malloc, allocating multiple blocks of memory: Calloc. Releasing the used space: Free, Altering the size of block: Realloc

Unit 5: Files
Introduction to files, Defining and opening file, closing file, I/O operation on files, error handling during I/O operations, Random Access to files and command line arguments.
Unit 6: Preprocessor

The C Preprocessor - How it works, conditional and unconditional directives, preprocessor commands, Introduction of Macros.

Text Books

2. Computer Fundamental & C programming by J.B.Dixit ; University Science Press
3. Fundamentals of Computer by V Rajaraman; Prentice Hall of India Pvt. Ltd., New Delhi

Reference Books

1. Let Us C by Yaswant Kanetkar : BPB Publication
2. Computer Fundamental & C programming by E. Balaguruswamy; MGH
3. Computers Today by SK Basandara, Galgotia publication Pvt ltd. Daryaganj, New Delhi

LAB:

Note: At least ten experiments are to be performed during the semester. At least eight experiments should be performed from the list of experiments. Two experiments may either be performed from the given list of experiments or may be designed by the concern faculty in consultation with H.O.D as per the scope of syllabus.

Objective: To understand the concepts of programming.

List of Experiments:

Note: C/C++ can be used to implement the following programs.
1- Program to concatenate the strings into single string without using library functions.
2- Program to copy one string into another without using library functions.
3- Program to reverse a string without using library functions.
4- Program to compare two strings without using library functions.
5- Program to demonstrate the implementation of pointers and its arithmetic’s.
6- Program to demonstrate the implementation of array of pointers.
7- Program to demonstrate the implementation of pointers and character strings.
8- Program to demonstrate the implementation of pointers as function arguments.
9- Program to demonstrate the use of malloc() and calloc().
10- Program to demonstrate the file handling e.g. file copy etc.
11- Program to demonstrate the use macros.
## Annexure-2

### Semester-I

<table>
<thead>
<tr>
<th>S. No</th>
<th>Course Title</th>
<th>Paper Code</th>
<th>Teaching Schedule</th>
<th>Total</th>
<th>Credit</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Object Oriented Programming using C++</td>
<td>CSL203</td>
<td>3 L 1 T 2 P</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>Data Structure &amp; Applications</td>
<td>CSL205</td>
<td>3 L 1 T 2 P</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>Discrete Structures</td>
<td>CSL207</td>
<td>3 L 1 T 0 P</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>Digital Electronics</td>
<td>ECL203</td>
<td>3 L 1 T 2 P</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td>Managerial Economics</td>
<td>MSL503</td>
<td>3 L 0 T 0 P</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>6</td>
<td>Applied Mathematics-III</td>
<td>AHL211</td>
<td>3 L 1 T 0 P</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>7</td>
<td>Professional Communication – III</td>
<td>AHP201</td>
<td>1 L 0 T 2 P</td>
<td>3</td>
<td>NC</td>
</tr>
<tr>
<td></td>
<td>Total</td>
<td></td>
<td>19 L 5 T 8 P</td>
<td>32</td>
<td>26</td>
</tr>
</tbody>
</table>

### Semester-IV

<table>
<thead>
<tr>
<th>S. No</th>
<th>Course Title</th>
<th>Paper Code</th>
<th>Teaching Schedule</th>
<th>Total</th>
<th>Credit</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Computer Architecture &amp;Organization</td>
<td>CSL 202</td>
<td>3 L 1 T 0 P</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>Database Admin Concepts &amp; Management</td>
<td>CSL 204</td>
<td>3 L 1 T 2 P</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>Software Engineering Concepts</td>
<td>CSL 206</td>
<td>3 L 0 T 0 P</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>Core Java</td>
<td>CSL 208</td>
<td>3 L 1 T 2 P</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td>Internet &amp;Web Technology</td>
<td>CSL 212</td>
<td>3 L 0 T 2 P</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>Programming Language</td>
<td>CSL 214</td>
<td>3 L 0 T 0 P</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>7</td>
<td>VAC-I</td>
<td>CSV202</td>
<td>0 L 0 T 2 P</td>
<td>2</td>
<td>NC</td>
</tr>
<tr>
<td></td>
<td>Total</td>
<td></td>
<td>18 L 3 T 8 P</td>
<td>29</td>
<td>24</td>
</tr>
</tbody>
</table>
Objective
This subject focuses better understanding and deeper knowledge of the Object and its uses. It also focuses on various object oriented programming concepts like inheritance, polymorphism, etc.

Theory
**Note:** Total five questions are to be attempted from Sections A & B.

**SECTION – A**

**Unit 1: Introduction to C++ and Object oriented Concepts**

C++ Standard Library, Basics of a Typical C++ Environment, Pre-processors Directives, Illustrative Simple C++ Programs. Header Files and Namespaces, library files. Introduction to Objects and Object Oriented Programming, Encapsulation (Information Hiding), Access Modifiers: Controlling access to a class, method, or variable (public, protected, private, package), Other Modifiers, Polymorphism: Overloading,, Inheritance, Overriding Methods, Abstract Classes, Reusability, Class’s Behaviors.

**Unit 2: Classes and Data Abstraction:**

Introduction, Structure Definitions, Accessing Members of Structures, Class Scope and Accessing Class Members, Separating Interface from Implementation, Controlling Access Function And Utility Functions, Initializing Class Objects: Constructors, Using Default Arguments With Constructors, Using Destructors, Classes : Const(Constant) Object And Const Member Functions, Object as Member of Classes, Friend Function and Friend Classes, Using This Pointer, Dynamic Memory Allocation with New and Delete, Static Class Members, Container Classes And Integrators, Proxy Classes, Function overloading.

**Unit 3: Operator Overloading:**

Fundamentals of Operator Overloading, Restrictions On Operators Overloading, Operator Functions as Class Members vs. as Friend Functions, Overloading, <<, >> Overloading Unary Operators, Overloading Binary Operators.

**SECTION – B**

**Unit 4: Inheritance and Polymorphism:**

Introduction to Inheritance, Base Classes And Derived Classes, Protected Members, Casting Base- Class Pointers to Derived- Class Pointers, Using Member Functions, Overriding Base – Class Members in a Derived Class, Public, Protected and Private Inheritance, Using Constructors and Destructors in derived Classes, Implicit Derived –Class Object To Base- Class Object Conversion, Composition Vs. Inheritance. Introduction to Virtual Functions, Abstract Base Classes And Concrete Classes, Polymorphism, New Classes And Dynamic Binding, Virtual Destructors, Polymorphism, Dynamic Binding.
Unit 5: Files and I/O Streams


Unit 6: Templates and Exception Handling:

Function Templates, Overloading Template Functions, Class Template, Class Templates and Non-Type Parameters, Templates and Inheritance, Templates and Friends, Templates and Static Members.Introduction, Basics of C++ Exception Handling: Try Throw, Catch, Throwing an Exception, Catching an Exception, Rethrowing an Exception, Exception specifications, Processing Unexpected Exceptions, Stack Unwinding, Constructors, Destructors and Exception Handling, Exceptions and Inheritance.

Text Books

3- Programming with C++ By D Ravichandran, 2003, T.M.H

Reference Books

2- Computing Concepts with C++ Essentials by Horstmann, 2003, John Wiley,
3- The Complete Reference in C++ By Herbert Schildt, 2002, TMH

LAB:

Note: At least ten experiments are to be performed during the semester. At least eight experiments should be performed from the list of experiments. Two experiments may either be performed from the given list of experiments or may be designed by the concern faculty in consultation with H.O.D as per the scope of syllabus.

Objective: To understand the concepts of object oriented programming techniques.

List of Experiments:

1- WAP :-Raising a number n to a power p is the same as multiplying n by itself p times. Write a function called power ( ) that takes a double value for n and an int value for p, and returns the result as double value. Use a default argument of 2 for p, so that if this argument is omitted, the number will be squared. Write a main ( ) function that gets values from the user to test this function.
2- WAP :-A point on the two dimensional plane can be represented by two numbers: an X coordinateand a Y coordinate. For example, (4,5) represents a point 4 units to the right of the origin along the X axis and 5 units up the Y axis. The sum of two points can be defined as a new point whose X coordinate is the sum of the X coordinates of the points and whose Y coordinate is the sum of their Y coordinates. Write a program that uses a structure called point to model a point. Define three points, and have the user input values to two of them. Than set the third point equal to the sum of the other two, and display the value of the new point.

3- WAP :-Create the equivalent of a four function calculator. The program should request the user to enter a number, an operator, and another number. It should then carry out the specified arithmetical operation: adding, subtracting, multiplying, or dividing the two numbers. (It should use a switch statement to select the operation). Finally it should display the result. When it finishes the calculation, the program should ask if the user wants to do another calculation. The response can be ‘Y’ or ‘N’.

4- WAP :-A phone number, such as (212) 767-8900, can be thought of as having three parts: the area code (212), the exchange (767) and the number (8900). Write a program that uses a structure to store these three parts of a phone number separately. Call the structure phone. Create two structure variables of type phone. Initialize one, and have the user input a number for the other one. Then display both numbers.

5- WAP :-Create two classes DM and DB which store the value of distances. DM stores distances in metres and centimeters and DB in feet and inches. Write a program that can read values for the class objects and add one object of DM with another object of DB. Use a friend function to carry out the addition operation. The object that stores the results maybe a DM object or DB object, depending on the units in which the results are required. The display should be in the format of feet and inches or metres and centimetres depending on the object on display.

6- WAP :-Create a class rational which represents a numerical value by two double values NUMERATOR & DENOMINATOR. Include the following public member Functions:
   a. constructor with no arguments (default).
   b. constructor with two arguments.
   c. void reduce( ) that reduces the rational number by eliminating the highest common factor .
   d. Overload + operator to add two rational number.
   e. Overload >> operator to enable input through cin.
   f. Overload << operator to enable output through cout
Write a main ( ) to test all the functions in the class.

7- WAP :-Consider the following class definition
   class father {
      protected : int age;
      public;
      father (int x) {age = x;}
   }
virtual void iam ( )
{
    cout << “I AM THE FATHER, my age is :” << age << endl;
}

Derive the two classes son and daughter from the above class and for each, define iam ( ) to write our similar but appropriate messages. You should also define suitable constructors for these classes.

Now, write a main ( ) that creates objects of the three classes and then calls iam ( ) for them. Declare pointer to father. Successively, assign addresses of objects of the two derived classes to this pointer and in each case, call iam ( ) through the pointer to demonstrate polymorphism in action.

8- WAP :-Write a program that creates a binary file by reading the data for the students from the terminal. The data of each student consist of roll no., name (a string of 30 or lesser no. of characters) and marks.

9- WAP :-A hospital wants to create a database regarding its indoor patients. The information to store include
   a) Name of the patient
   b) Date of admission
   c) Disease
   d) Date of discharge
Create a structure to store the date (year, month and date as its members). Create a base class to store the above information. The member function should include functions to enter information and display a list of all the patients in the database. Create a derived class to store the age of the patients. List the information about all the to store the age of the patients. List the information about all the pediatric patients (less than twelve years in age).

10- WAP :-Make a class Employee with a name and salary. Make a class Manager inherit from Employee. Add an instance variable, named department, of type string. Supply a method to toString that prints the manager’s name, department and salary. Make a class Executive inherit from Manager. Supply a method to String that prints the string “Executive” followed by the information stored in the Manager superclass object. Supply a test program that tests these classes and methods.

11- WAP :- Imagine a tollbooth with a class called toll Booth. The two data items are a type unsigned int to hold the total number of cars, and a type double to hold the total amount of money collected. A constructor initializes both these to 0. A member function called payingCar ( ) increments the car total and adds 0.50 to the cash total. Another function, called nopayCar ( ), increments the car total but adds nothing to the cash total. Finally, a member function called displays the two totals.
Include a program to test this class. This program should allow the user to push one key to count a paying car, and another to count a nonpaying car. Pushing the ESC key should cause the program to print out the total cars and total cash and then exit.

12- WAP :: Write a function called reversit ( ) that reverses a string (an array of char). Use a for loop that swaps the first and last characters, then the second and next to last characters and so on. The string should be passed to reversit ( ) as an argument.
Write a program to exercise reversit ( ). The program should get a string from the user, call reversit ( ), and print out the result. Use an input method that allows embedded blanks. Test the program with Napoleon’s famous phrase, “Able was I ere I saw Elba”.

13- WAP :: Create some objects of the string class, and put them in a Deque—some at the head of the Deque and some at the tail. Display the contents of the Deque using the forEach ( ) function and a user written display function. Then search the Deque for a particular string, using the firstThat ( ) function and display any strings that match. Finally remove all the items from the Deque using the getLeft ( ) function and display each item. Notice the order in which the items are displayed: Using getLeft ( ), those inserted on the left (head) of the Deque are removed in “last in first out” order while those put on the right side are removed in “first in first out” order. The opposite would be true if getRight ( ) were used

14- WAP :: Create a base class called shape. Use this class to store two double type values that could be used to compute the area of figures. Derive two specific classes called triangle and rectangle from the base shape. Add to the base class, a member function get_data ( ) to initialize base class data members and another member function display_area ( ) to compute and display the area of figures. Make display_area ( ) as a virtual function and redefine this function in the derived classes to suit their requirements.
Using these three classes, design a program that will accept dimensions of a triangle or a rectangle interactively and display the area.

Remember the two values given as input will be treated as lengths of two sides in the case of rectangles and as base and height in the case of triangles and used as follows:

\[
\text{Area of rectangle} = x \times y \\
\text{Area of triangle} = \frac{1}{2} \times x \times y
\]
Objective:
This subject discusses different data structures to represent real world problems and to study various ways to design algorithms to solve the problems.

Theory
Note: Total five questions are to be attempted from Sections A & B.

SECTION – A

Unit 1: Introduction to Data Structures:
Definition of data structures and abstract data types, Static and Dynamic implementations, Elementary Data Organization, Data Structure operations, Algorithms, Time and space complexity of algorithms. Arrays: Representation of single and multidimensional arrays; sparse arrays - lower and upper triangular matrices.

Unit 2: Searching and Sorting

Unit 3: Stack and Queues
Stacks: Introduction and primitive operations on stack; Application of stack: Stack frames, Conversion of Infix to Prefix and Postfix Expressions, Evaluation of postfix expression using stack. Queues: Introduction, Array and linked representation of queues, primitive operation on queues, D-queues and priority queues

SECTION – B

Unit 4: Linked List
Representation and Implementation of Singly Linked Lists, Two-way Header List, Traversing and Searching of Linked List, Insertion and deletion to/from Linked Lists, Insertion and deletion Algorithms, Doubly linked list(Insertion and deletion, Traversing and Searching), Circular Linked List(Insertion and deletion, Traversing and Searching).

Unit 5: Tree

**Unit 6: Graph**


**Text Book:**


**Reference Books**


**LAB:**

**Note:** At least ten experiments are to be performed during the semester. At least eight experiments should be performed from the list of experiments. Two experiments may either be performed from the given list of experiments or may be designed by the concern faculty in consultation with H.O.D as per the scope of syllabus.

**Objective:** To understand the concepts of various data structures, searching, and sorting techniques.

**List of Experiments:**

Note: C/C++ can be used to implement the following programs.
1-Implementation of 1D and 2D array using static memory allocation.
2-Implementation of 1D and 2D array using dynamic memory allocation with the help of pointer.
3-Implementation of sparse matrices.
4-Implementation of linear and binary search.
5-Implementation of Bubble sort and Quick
6-Implementation of Heap sort
7-Implementation of Merge sort
8-Implementation of PUSH and POP operation into STACK
9-Implementation of Queue and priority queue
10-Implementation of creation, insertion, deletion, and searching operation in linked list.
11-Implementation of Binary Search tree and its operations
12-Implementation of Graph traversal.
13-Implementation of Minimum spanning tree in graph.
Objective
This subject enhances the ability to formulate and solve applied problems, to analyze and interpret algorithms and functions and to use them effectively. The goal is to make a student learn how to think about discrete mathematical models.

Theory
Note: Total five questions are to be attempted from Sections A & B.

SECTION – A

Unit 1: Set Theory
Introduction to set theory, Set operations, Algebra of sets, Duality, Finite and Infinite sets, Classes of sets, Power Sets, Multi sets, Cartesian Product, Representation of relations, Types of relation, Equivalence relations and partitions, Partial ordering relations, inverse Relations Function and its types, Composition of function and relations, Cardinality, Principle of inclusion and exclusion, Pigeon hole principle.

Unit 2: Propositional Calculus:
Introduction to propositional Calculus: Basic operations: AND(^), OR(\lor), NOT(\lnot), Truth value of a compound statement, propositions, tautologies, contradictions.

Unit 3: Techniques of Counting
Permutations with and without repetition, Combination, Polynomials and their evaluation.

SECTION – B

Unit 4: Recursion and recurrence Relation:
partial fractions, linear recurrence relation with constant coefficients, Homogeneous solutions, Particular solutions, Total solution of a recurrence relation using generating functions.

Unit 5: : Algebraic Structure
Properties of Algebraic Structures, Definition and examples of a monoid, Semigroup, Groups, Subgroups, Abelian Group, rings, Homomorphism, Isomorphism and Automorphism, Normal subgroups, Cyclic groups, Integral domain and fields, Cosets, Lagrange’s theorem
Unit 6: Graphs and Trees

Introduction to graphs, Directed and Undirected graphs, Homomorphic and Isomorphic graphs, Subgraphs, Cut points and Bridges, Multigraph and Weighted graph, Paths and circuits, walk, Shortest path in weighted graphs, Eurelian path and circuits, Hamilton paths and circuits, Planar graphs, Euler’s formula, Trees, Spanning trees, Binary trees and its traversals

Text Book:


Reference Books:


3- Applied Discrete Structures for Computer Science, Doerr and Levasseur, (Chicago: 1985,SRA

4- Discrete Mathematics and Structures by Satinder Bal Gupta, University science press.
Objective
The objective of the course is to get the knowledge about the designing principles of different digital electronics circuits and to study their applications.

Theory
Note: Total five questions are to be attempted from Sections A & B.

SECTION – A

UNIT 1: Digital system and binary numbers
Introduction of digital system, Signed binary numbers, cyclic codes, BCD codes, Excess-3 code, error detecting and correcting codes, hamming codes, floating point representation of number, Boolean algebra: Representation of values and complements, AND, OR, NOT operators, D’Morgans theorem-simplifying expression simple problems

UNIT 2: Gate-level minimization
Truth table of all logic gate’s, universal gate, NAND and NOR implementation The K-map method up to five variable, don’t care conditions, POS & SOP simplification, Quine Mc-CLUSky method (Tabular method).

UNIT 3: Combinational Logic
Combinational circuits, analysis procedure, design procedure, binary adder-subtractor, decimal adder, Parity Generator and checker ,binary multiplier, magnitude comparator, decoders, encoders, multiplexers, De-Multiplexer

SECTION – B

UNIT 4: Sequential Logic
Sequential circuits, storage elements: latches, flip flops, analysis of clocked sequential circuits, state diagram, state reduction and assignments, design procedure for clocked sequential circuit from state diagram Registers and counters: Shift registers; ripple counter, synchronous counter.

UNIT 5: Memory and Programmable Logic
Introduction PLD’s types of PLD’s, RAM & ROM, PLA, and PAL, FPGA and CPLD, sequential logic design using PLA& PAL. Design at the register transfer level: Introduction ASMs, ASM chart, and design example with multiplexers.

UNIT 6: Asynchronous Sequential Logic
Introduction of ASL circuits, types of ASL circuits, Analysis procedure, circuit with latches, design procedure, reduction of state and flow table, Primitive table, problems in Asynchronous circuits ,race Free State assignment, hazards, Design of hazard-free switching circuits

Text Books:
Reference Books:

LAB:
Note: At least ten experiments are to be performed during the semester. At least eight experiments should be performed from the list of experiments. Two experiments may either be performed from the given list of experiments or may be designed by the concern faculty in consultation with H.O.D as per the scope of syllabus.

Objective: To understand the digital logic and create various systems by using these logics

List of Experiments:

1. Introduction to digital electronics lab- nomenclature of digital ICs, specifications, study of the data sheet, concept of Vcc and ground, verification of the truth tables of logic gates using TTL ICs.
2. Implementation of the given Boolean function using logic gates in both SOP and POS forms.
3. Verification of state tables of RS, JK, T and D flip-flops using NAND & NOR gates.
4. Implementation and verification of Decoder/De-multiplexer and Encoder using logic gates.
5. Implementation of 4x1 multiplexer using logic gates.
6. Design, and verify the 4-bit synchronous counter.
7. Design, and verify the 4-bit asynchronous counter.
8. Implement and verification of the truth table of Half-Adder and Full-Adder.
9. To study and implement the 4- Bit Magnitude comparator.
10. Mini Project.
Objective
The paper seeks to equip the students with the analytical tools of Economics and apply the same to rational managerial decision-making. It further seeks to develop economic way of thinking in dealing with practical business problems and challenges.

Theory
Note: Total five questions are to be attempted from Sections A & B.

SECTION – A


SECTION – B


Unit 5: Theory and Behavior of firm – Different market structure and their characteristics: Perfect competition, Monopolistic competition, Monopoly and Oligopoly.

Unit 6: Indian Economy – Nature of Indian Economy, Privation: meaning, Merits and Demerits, Globalisation: meaning, Merits and Demerits.

Text Books

1. Dwivedi, D.N., Managerial Economics, Vikas publishing house pvt ltd, New Delhi, Ed 2006
Reference Books

1- Hirschey, Mark, Managerial Economics, Thomson Learning, Bangalore


4- Salvatore, Dominick, Managerial Economics in a Global Economy, Thomson Learning, Hyderabad
Objective
This subject focuses Fourier series, complex analysis, linear, transportation, and assignment problems, which are used in various computer applications.

Theory
Note: Total five questions are to be attempted from Sections A & B.

SECTION – A

Unit 1:
FOURIER SERIES: Euler’s formulae, conditions for a Fourier expansion, change of interval, Fourier expansion of odd and even functions, Fourier expansion of square wave, rectangular wave, saw-toothed wave, half and full rectified wave, half range sine and cosine series.

Unit 2:
COMPLEX ANALYSIS: Functions of Complex Variable, Definition, Exponential function, Trigonometric and Hyperbolic functions, Logarithmic functions. Limit and Continuity of a function, Differentiability and analyticity Cauchy-Riemann equations, necessary and sufficient conditions for a function to be analytic (without proof), polar form of the Cauchy-Riemann equations. Harmonic functions, Integration of complex functions. Cauchy-Integral theorem and formula.

Unit 3:
POWER SERIES AND CONTOUR INTEGRATION: Power series, radius and circle of convergence, Taylor’s Malaren’s and Laurent’s series. Zeroes and Singularities of complex functions, Residues. Evaluation of real integrals using residues (around unit and semi circle only).

SECTION – B

Unit 4:
LINEAR PROGRAMMING PROBLEMS: Formulation, Solving linear programming problems using (i) Graphical method (ii) Simplex method (iii) Big –M method (iv) Dual simplex method (v) Two phase method.

Unit 5:
TRANSPORTATION PROBLEM: Formulation, Basic feasible solutions, optimum solution by u-v method, unbalanced and degenerate problems.
Unit 6:

ASSIGNMENT PROBLEM: Formulation, Solution by Hungarian method, unbalanced problem, case of maximization, travelling salesman and crew assignment problems.

Text Books

2- Operation Research by S.D.Sharma.

Reference Books

1- Higher Engineering Mathematics by B.S. Grewal
UNIT-1: PHONETICS
   i. Recapitulation of Consonant and Vowel Sounds
   ii. Concept of Phonemes and Allophones
   iii. Concept of Minimal Pairs
   iv. Differences in pronunciation between words with
       • /f/ and /P/ sounds
       • /z/ and /dз/ sounds
       • /S/ and /ʃ/ sounds
       • /v/ and /w/ sounds
       • /tʃ/ and /ʃ/ sounds
   v. Words commonly mispronounced

UNIT-2: READING PRACTICE
   i. Reading newspaper articles
   ii. Reading magazine articles

UNIT-3: ORAL PRACTICE
   i. Self Introduction
   ii. Introducing others
   iii. Speeches with prior preparation
   iv. Extempore Speeches
   v. Free Speeches

UNIT-4: STUDY SKILLS
   i. Looking up a dictionary
   ii. Learning pronunciation from a dictionary
   iii. Filling up of examination forms
   iv. Spell check
### Objective

Computer architecture is a key component of computer engineering. The computer Architecture and organization is concerned with the structure and behavior of digital computers. The main objective of this subject is to understand the overall basic computer hardware structure, including the peripheral devices.

### Theory

**Note:** Total five questions are to be attempted from Sections A & B.

#### SECTION – A

**Unit 1**

Review of Basic Computer Architecture and Microprocessors; Von Neumann architecture: principles, instruction sets, instruction format, addressing modes.

**Unit 2**

CISC versus RISC architectures, Storage system and their technology: memory hierarchy, main memory organization and operations, cycle time, bandwidth and interleaving; cache memory: addressing mapping, block size, replacement and store policy.

**Unit 3**

virtual memory: page table, TLB; I/O fundamentals: handshaking, buffering, programmed I/O, interrupts-driven I/O; Buses: types, bus protocols, arbitration,

#### SECTION – B

**Unit 4**

Direct Memory Access, Pipelining: principles, Instruction pipelines, Pipelines difficulties and solutions, Introduction to SIMD, MIMD.

**Unit 5**

subroutine call and return mechanism; Control unit: hardwired, micro-programmed, micro instruction mapping, micro program sequencer.

**Unit 6**

Architecture of 8086, Instruction set of 8086, Assembly/machine language programming (8086).
Text Book:


Reference Books

MVN University, Haryana

<table>
<thead>
<tr>
<th>CSL 204</th>
<th>Database Admin Concepts and Management</th>
<th>L</th>
<th>T</th>
<th>P</th>
<th>Cr</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>5</td>
</tr>
</tbody>
</table>

**Objective**

the design methodology for databases and verifying their structural correctness implementing databases and applications software primarily in the relational model using querying languages, primarily SQL, and other database supporting software applying the theory behind various database models and query languages

**Theory**

*Note:* Total five questions are to be attempted from Sections A & B.

**SECTION A**

**Unit 1:**
Introduction to database Management System; Various views of data, data Models, Introduction to Database Languages. Advantages of DBMS over file processing systems. Introduction to Client/Server architecture, Three levels architecture of Database Systems, E-R Diagram (Entity Relationship), mapping Constraints, Keys, Reduction of E-R diagram into tables.

**Unit 2:**
Relational Model, Relational Algebra & various operations

**Unit 3:**

**SECTION B**

**Unit 4**
Sequential Files, index sequential files, direct files, Hashing, B-trees Index files.

**Unit 5**
Introduction to transaction, properties of transaction and life cycle of transaction, Introduction to Concurrency, Why concurrency Needed, Concurrency control Techniques (Two phase locking protocol, Timestamp based locking protocol, Validation Based protocols) and Recovery Concept.

**Unit 6**
Introduction to Database Administrator , Responsibility of Database Administrator, type of Database Administrator.

**Text Books:**


Reference Books:

2- Introduction to Database Management System by Satinder Bal Gupta and Aditya Mittal

LAB:
Note: At least ten experiments are to be performed during the semester. At least eight experiments should be performed from the list of experiments. Two experiments may either be performed from the given list of experiments or may be designed by the concern faculty in consultation with H.O.D as per the scope of syllabus.

Objective: To understand the concepts of database, operation of database and various queries.

List of Experiments:
Note: Create a database and write the programs to carry out the following operation:

Create tables and specify the Questionnaires in SQL

1- Add a record in the database
2- Delete a record in the database
3- Modify the record in the database
4- To implement the restrictions on the table
5- List all the records of database in ascending order.
6- To implement the structure of the table.
7- To Implement Oracle function.
8- To implement the concept of grouping of Data
9- To implement the concept of Joins
Objective
This subject deals with the designing and testing of software. Various models and techniques are discussed in this subject to improve performance of software.

Theory
Note: Total five questions are to be attempted from Sections A & B.

SECTION – A

Software life cycle models: Waterfall, Evolutionary and Spiral models, Selection of a Life Cycle Model


SECTION – B

Software Reliability: Basic concepts, software quality, software reliability models

Unit 5: Software Testing: A Strategic Approach of Software Testing, some Terminologies: error, fault, bugs, mistake and failure, test cases, test, test suite, verification and validation, alpha, beta and acceptance testing, functional testing, structural testing, level of testing, The art of debugging.

Text Books:


Reference Books:

Objective
This subject provides the knowledge of basic Java language. It also discusses threads, packages, and applet, and other advanced features used in language.

Theory
Note: Total five questions are to be attempted from Sections A & B.

SECTION – A

Unit 1: Introduction to java
Features of JAVA, Java virtual machine, Java runtime environment, Variables and data types, Conditional and looping constructs, Fields and Methods, objects & classes, Constructors, Overloading methods, Garbage collection, Nested classes, Array, String, StringBuffer, String Builder & Vectors, Packages, Access Modifiers, Enumerations, Auto boxing, and Annotations.

Unit 2: Inheritance & Exception Handling
Use and Benefits of Inheritance in OOP, Types of Inheritance in Java, Inheriting Data Members and Methods, Role of Constructors in inheritance, Overriding methods, Polymorphism, Making methods and classes final, Abstract classes and methods, use of super & this, Packages & Interfaces. Exceptions Overview, Exceptions & Errors, Types of Exception, use of try, catch, finally, throw, throws in Exception Handling, In-built and User Defined Exceptions, Checked and Un-Checked Exceptions, Assertions.

Unit 3: Threads & Reflection
Needs of Multi-Threaded Programming, Processes & threads, Thread Life-Cycle, Thread class & Runnable interface, Thread Priorities, Synchronizing Threads, Inter Communication of Threads, Reflection: introduction, Use of reflection, Retrieving Class Objects, Discovering Class Members

SECTION – B

Unit 4: Collection Framework & Generics

Unit 5: GUI & Java Applets

Unit 6: Java Streams, Files & I/O
Input Streams, Output Streams, Reading console input, writing console output, reading and writing files, object serialization & deserialization, Path class, File class, creating directories & reading directories.

**Text Book**

1- JAVA: The Complete Reference, Herbert Schildt  
2- Core JavaTM 2, Volume II-Advanced Features, 7th Edition by Cay Horetmann, Gary Cornelll Pearson Publisher,2004

**Reference Books**

1- Professional Java Programming by Brett Spell, WROX Publication  
3- Advanced Java, Gajendra Gupta, Firewall Media

**LAB:**  
**Note:** At least ten experiments are to be performed during the semester. At least eight experiments should be performed from the list of experiments. Two experiments may either be performed from the given list of experiments or may be designed by the concern faculty in consultation with H.O.D as per the scope of syllabus.

**Objective:** To understand the concepts of core java.

**List of Experiments:**
Note: JAVA must be used to implement the following programs.

1- Write a program to find the area of rectangle.  
2- Write a program to print n terms of the Fibonacci series.  
3- Write a program to find largest and smallest number in an array.  
4- Write a program to multiply two matrices.  
5- Write a program to implement constructor overloading.  
6- Write a program to implement different types of inheritance.  
7- Write a program to implement method overriding.  
8- Write a program to implement multiple inheritance using interfaces.  
9- Write a program to implement run time polymorphism.  
10- Write a program to catch more than two exceptions.  
11- Write a program to create a user defined exception”NegativeAgeException” that will be thrown by program if input age given by user is negative.  
12- Write a java program generating two threads-one for generating even numbers and one for generating odd numbers.  
13- Write a program to print the information of a class using reflection.  
14- Write a program to develop a producer –consumer problem using thread.  
15- Write a program using ArrayList and LinkedList collections.  
16- Write a program using TreeSet and HashSet collections.  
17- Write a program using HashMap and TreeMap collections.  
18- Write a program to develop a simple calculator.  
19- Write a program to read data from a file and to write data into a file.  
20- Write a program to create directory and reading from directories.
Objective

Students will get an introduction about various Scripting Languages. Students will be provided with an up-to-date survey of developments in Web Development. Enable the students to know techniques involved to support real-time Software development.

Theory

Note: Total five questions are to be attempted from Sections A & B.

Section A

Unit 1. The Internet:

History of the Internet and Application, Modes of Connecting to Internet, Internet Service Providers (ISPs), Working of Internet, Internet Congestion, Domain Name Space, Internet address, IPv4, IPv6, E-Mail System, Protocols of Email, -mail management, Mime types, Newsgroups, mailing lists, chat rooms, Internet tool.

Unit 2 WWW:


Unit 3 HTML:

History and introduction of HTML, HTML Basic tag, image, Hyperlink, formatting of page, table, list, frame, form, Meta tags, Dynamic Hypertext Markup Language, HTML tool: Front page.

Section B

Unit 4: Cascading Style Sheet:


Unit 5: Scripting languages:

Unit 6: Web Related other concepts:

History and introduction, Document type definition, XML Schemas, Document Object model, Presenting XML, Using XML Processors: DOM and SAX. Introduction to server side scripting language: ASP, JSP, PHP.

Text Book

2- HTML & XHTML: The Definitive Guide by Chuck Musciano, Bill Kennedy, 2000, 4th Edi. Reference books:
3- All-in-one Desk Reference for Dummies by Andy Harris

Reference Books:

1- Complete idiots guide to java script,. Aron Weiss, QUE, 1997
2- Internet and Web Technologies – Raj Kamal 2002, T.M.H

LAB:
Note: At least ten experiments are to be performed during the semester. At least eight experiments should be performed from the list of experiments. Two experiments may either be performed from the given list of experiments or may be designed by the concern faculty in consultation with H.O.D as per the scope of syllabus.

Objective: To understand the concepts of various web technologies.

List of Experiments:

1- Sending and Receiving Mail
2- Remote Login using Telnet
3- To demonstrate the basic HTML tag.
4- To demonstrate the use of list using HTML tag.
5- To demonstrate the use of links using HTML tags.
6- To demonstrate the use of image links using HTML tags.
7- To demonstrate the use of different shapes using HTML tags.
8- To demonstrate the use of table using HTML tag.
9- To demonstrate the use of frame using HTML tag.
10- To demonstrate the use of forms using HTML tag.
11- Create a web page using internal CSS
12- Create a web page using external CSS
13- Create a web page using embedded CSS
14- To demonstrate the use of prompt, Alert and confirm tags using java script.
15- JavaScript. Create a web page to access forms using
16- Create a web page to validate an email id using JavaScript.

List of Value added Experiments:

1- Create a website for your college.
2- Create a website for newspaper agency.
Objective
Programming language theory liberates students from the tar pit of personal opinion, and elevates them to the level of respectable scientific discourse.

Theory
Note: Total five questions are to be attempted from Sections A & B.

SECTION – A

UNIT 1: Introduction

UNIT 2: Data Types
Elementary data types – data objects, variable & constants, data types, Specification & Implementation of elementary data types, Declarations, type checking & type conversions, Assignment & initialization, Numeric data types, enumerations, Booleans & characters. Structured data objects & data types, specification & implementation of structured data types, Declaration & type checking of data structure, vector & arrays.

UNIT 3: Subprograms and Programmer Define data types
Encapsulation by Subprograms –Subprogram as abstract operations, Subprogram definition and invocation, Subprogram definition as data objects, Generic subprogram. Evolution of data type concept, abstraction, abstract data types, encapsulation & information hiding, Inheritance

Section B

Unit 4 : Sequence Control
Implicit & explicit sequence control, sequence control within expressions, sequence control within statement, Subprogram sequence control: simple call return, recursive subprograms, Exception & exception handlers, co routines.

UNIT 5 : Storage Management
Major run time elements requiring storage, programmer and system controlled storage Management & phases, Static storage management, Stack based storage management, Heap Storage management, variable & fixed size elements.

UNIT 6: Programming Languages
MVN University, Haryana

Introduction to procedural, nonprocedural, structured, functional and object oriented programming language, Comparison of C & C++ programming languages.

**Text Books**

1- Concepts of Programming Languages Robert .W. Sebesta 6/e, Pearson Education.
3- Programming Languages Design and Implementation “ Pratt and Zelkowitz, Fourth Edition
   PHI/Pearson Education

**Reference Books**

1- Programming languages Ghezzi, 3/e, John Wiley
2- Programming languages Watt, Wiley Dreamtech
3- LISP Patric Henry Winston and Paul Horn Pearson Education
Any technical subject can be taught which is suitable according to current industrial need and which is not being taught as part of current curriculum, decided by a departmental committee which headed by the head of the department.
### Semester-V

<table>
<thead>
<tr>
<th>S. No</th>
<th>Course Title</th>
<th>Paper Code</th>
<th>Teaching Schedule</th>
<th>Total</th>
<th>Credit</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>L</td>
<td>T</td>
<td>P</td>
</tr>
<tr>
<td>1</td>
<td>Operating System</td>
<td>CSL301</td>
<td>3</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>Multimedia &amp; Animation</td>
<td>CSL303</td>
<td>3</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>FL &amp; Automata Theory</td>
<td>CSL305</td>
<td>3</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>Algorithm Design &amp; Analysis</td>
<td>CSL307</td>
<td>3</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>5</td>
<td>Microprocessor &amp; Interfacing</td>
<td>ECL307</td>
<td>3</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>Computer Graphics</td>
<td>CSL309</td>
<td>3</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>7</td>
<td>Practical Training - I</td>
<td>CST311</td>
<td>0</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>8</td>
<td>Professional Communication – IV</td>
<td>AHP301</td>
<td>1</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td></td>
<td>19</td>
<td>5</td>
<td>12</td>
</tr>
</tbody>
</table>

### Semester-VI

<table>
<thead>
<tr>
<th>S. No</th>
<th>Course Title</th>
<th>Paper Code</th>
<th>Teaching Schedule</th>
<th>Total</th>
<th>Credit</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>L</td>
<td>T</td>
<td>P</td>
</tr>
<tr>
<td>1</td>
<td>Distributed Operating System</td>
<td>CSL302</td>
<td>3</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>Artificial Intelligence</td>
<td>CSL304</td>
<td>3</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>Principles of Compiler Design</td>
<td>CSL306</td>
<td>3</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>Computer Networks</td>
<td>CSL308</td>
<td>3</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>5</td>
<td>Introduction to Cyber Security</td>
<td>CSL310</td>
<td>3</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td>Advanced java</td>
<td>CSL312</td>
<td>3</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>7</td>
<td>VAC –II</td>
<td>CSV302</td>
<td>0</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td></td>
<td>18</td>
<td>3</td>
<td>12</td>
</tr>
</tbody>
</table>
Objective
This subject focuses better understanding and deeper knowledge of Operating system.

Note: Question No 1 is compulsory and will be of short answer type from entire syllabus. Two questions are to be attempted out of three questions from each Section A & B.

Section-A
Unit 1: Introduction: Introduction to Operating System Concepts (including Multitasking, multiprogramming, multi user, Multithreading etc.). Types of Operating Systems: Batch operating system, Time-sharing systems, Distributed OS, Network OS, Real Time OS; Various Operating system services, architecture, System programs and calls.

Unit 2: Process Management: Process concept, process scheduling, operation on processes; CPU scheduling, scheduling criteria, CPU scheduling algorithms.

Unit 3: Memory Management: Logical & Physical Address Space, swapping, contiguous memory allocation, non-contiguous memory allocation paging and segmentation techniques, segmentation with paging; virtual memory management - Demand Paging & Page-Replacement Algorithms; Demand Segmentation.

Section-B
Unit 4: File System: Different types of files and their access methods, directory structures, various allocation methods, disk scheduling and management and its associated algorithms, Introduction to distributed file system.

Unit 5: Process-Synchronization & Deadlocks: Critical Section Problems, semaphores; methods for handling deadlocks-deadlock prevention, avoidance & detection; deadlock recovery.


Text Books:

Reference Books:
- Operating System By Milankovic, 1990, TMH
- Operating System Incorporating With Unix & Windows By Colin Ritche, 1974, TMH
LAB:

**Note:** At least ten experiments are to be performed during the semester. At least eight experiments should be performed from the list of experiments. Two experiments may either be performed from the given list of experiments or may be designed by the concern faculty in consultation with H.O.D as per the scope of syllabus.

**Objective:** To understand the command of UNIX and shell programming.

**List of Experiments:**

1. Introduction and architecture of Unix operating system.
2. Basic commands of Unix like cal, date, echo, passwd, ls, who, wc, tput, man, ps with their options.
3. Commands for creating files and directories.
5. Command for changing for the file permissions.
7. WAP for sum, multiplication, subtraction, division by using vi – editor.
8. WAP for searching a pattern from specified file using “grep” command.
9. WAP to add two no’s using function.
10. WAP to greet according to input.
11. WAP to print table of a given no.
12. WAP to print factorial of a no.
13. WAP to print a marksheet of a student.
14. WAP of calculator using switch case.
Objective:

This course fully covers the basics of multimedia technology.

THEORY:

Note: Question No 1 is compulsory and will be of short answer type from entire syllabus. Two questions are to be attempted out of three questions from each Section A & B.

Section-A


Stages of Multimedia Projects: Multimedia hardware, Memory & storage devices, Communication devices, Multimedia software's, presentation tools, tools for object generations, video, sound, image capturing,authoring tools, card and page based authoring tools.

Unit-2: Multimedia Building Blocks: Text, Sound MIDI, Digital Audio, audio file formats, MIDI under windows environment Audio& Video Capture.

Unit-3: Data Compression: Huffman Coding, Shannon Fano Algorithm, Huffman Algorithms, Adaptive Coding, Arithmetic Coding Higher Order Modelling, Finite Context Modelling, Dictionary based Compression, Sliding Window Compression, LZ77, LZW compression, Compression, Compression ratio lossless & lossy compression.

Section-B

Unit-4: Speech Compression & Synthesis
Digital Audio concepts, Sampling Variables, Loss less compression of sound, loss compression & silence compression.

Unit-5: Images &Video:
Multiple monitors, bitmaps, Vector drawing, lossy graphic compression, image file formats, JPEG Compression, Zig Zag Coding, Multimedia Database
Video : Video representation, Colors, Video Compression, MPEG standards, MHEG Standard Video Streaming on net, Video Conferencing, Multimedia Broadcast Services, Indexing and retrieval of Video Database.

Unit-6: Virtual Reality:
Applications of multimedia, intelligent multimedia system, desktop virtual reality, VR operating system, virtual environment displays and orientation making; visually coupled system requirements; intelligent VR software systems. Applications of environment in various fields.

References:
LAB:

Note: At least ten experiments should be performed from the given list of experiments. Five experiments may be designed by the concern faculty in consultation with H.O.D as per the scope of syllabus.

Objective: To understand the concepts of Multimedia.

List of Experiments:

1. Write a program to justify a text entered by the user on both the left and right hand side. For example, the text “An architect may have a graphics program to draw an entire building but be interested in only ground floor”, can be justified in 30 columns as shown below. An architect may have a Graphics program to draw an entire building but be interested in only ground floor.
2. Study the notes of a piano and stimulate them using the key board and store them in a file.
3. Write a program to read a paragraph and store it to a file name suggested by the author.
4. Devise a routine to produce the animation effect of a square transforming to a triangle and then to a circle.
5. Write a program to show a bitmap image on your computer screen.
6. Create a web page for a clothing company which contains all the details of that company and at-least five links to other web pages.
7. Write a program by which we can split mpeg video into smaller pieces for the purpose of sending it over the web or by small capacity floppy diskettes and then joining them at the destination.
8. Write a program to simulate the game of pool table.
9. Write a program to simulate the game Mine Sweeper.
10. Write a program to play “wave” or “midi” format sound files.
Objective
This subject focuses better understanding and deeper knowledge of languages and automata.

THEORY:
Note: Question No 1 is compulsory and will be of short answer type from entire syllabus. Two questions are to be attempted out of three questions from each Section A & B.

Section A
UNIT-1:
Finite Automata and Regular Expressions: Finite State Systems, Basic Definitions Non-Deterministic finite automata (NDFA), Deterministic finite automata (DFA), Equivalence of DFA and NDFA Conversion of NFA to DFA Finite automata with E-moves, Regular Expressions, Equivalence of finite automata and Regular Expressions, Regular expression conversion and vice versa.

UNIT-2:
Introduction to Machines: Concept of basic Machine, Properties and limitations of FSM. Moore and mealy Machines, Equivalence of Moore and Mealy machines.


UNIT-3:
Grammars & Context free grammar: Grammar, Chomsky hierarchies of grammars, derivation tree in CFG, ambiguous CFG, and Removal of useless Symbols, unit production and null production, Chomsky Normal Form (CNF), Griebach Normal Form (GNF).

Section B
UNIT-4:
Pushdown Automata: Description and definition, Instantaneous Description, Language of PDA, Acceptance by Final state, Acceptance by empty stack, Deterministic PDA, conversion: CFG to PDA and PDA to CFG,

UNIT-5:
Turing Machines: Deterministic and Non-Deterministic Turing Machines, Design of T.M, Universal TM, Church’s Thesis, Halting problem of T.M., PCP Problem, Recursive and recursively enumerable languages ,

UNIT-6:
Computability: Basic concepts, Primitive Recursive Functions.

Text Books:

Reference Books:
MVN University, Haryana

- Introduction to languages and the Theory of Computation by John C. Martin 2003, T.M.H.
- Introduction to formal Languages & Automata-Peter Linz, 2001, Narosa Publ
Objective: Today is the era of Computer. To discuss different data structures to represent real world problems and to study various ways to design algorithms to solve the problems.

Note: Question No 1 is compulsory and will be of short answer type from entire syllabus. Two questions are to be attempted out of three questions from each Section A & B.

Section-A

UNIT-I:
Introduction: Algorithm, Pseudocode for expressing algorithms, Performance analysis – space complexity, Time complexity, Asymptotic Notation – Big oh notation, Omega notation, Theta notation, sorting and searching algorithms and their analysis in terms of space and time complexity.

UNIT-II:
Divide and Conquer: General method, binary search, merge sort, quick sort, Strassen’s matrix multiplication algorithms and analysis of algorithms for these problems.

UNIT-III:
Greedy Method: General method, knapsack problem, job sequencing with dead lines, minimum spanning trees, single source paths and analysis of these problems.

Section -B

UNIT-IV:
Dynamic Programming: General method, optimal binary search trees, O/I knapsack, the traveling salesperson problem.

UNIT-V:
Back Tracking: General method, 8 queen’s problem, graph colouring, Hamiltonian cycles, analysis of these problems.
Branch and Bound: Method, O/I knapsack and traveling salesperson problem, efficiency considerations. Techniques for algebraic problems, some lower bounds on parallel computations.

UNIT-VI:
NP Hard and NP Complete Problems: Basic concepts, Cook’s theorem, NP hard graph and NP scheduling problems some simplified NP hard problems.

Text Books:
Introduction To Algorithms, Thomas H Cormen, Charles E Leiserson And Ronald L Rivest: 1990, TMH
Fundamental of Computer algorithms, Ellis Horowitz and Sartaj Sahni, 1978, Galgotia Publ.,
**Reference Books:**

Johan Wiley & Sons, Writing Efficient Programs, Bentley, J.L., PHI

Introduction to Design and Analysis of Algorithm, Goodman, S.E. & Hedetnieni, 1997, MGH.


**ALGORITHM DESIGN AND ANALYSIS LAB**

**Programming assignments on each algorithmic strategy:**

1. Divide and conquer method (quick sort, merge sort, Strassen’s matrix multiplication),

2. Greedy method (knapsack problem, job sequencing, optimal merge patterns, minimal spanning trees).

3. Dynamic programming (multistage graphs, OBST, 0/1 knapsack, traveling salesperson problem).


5. Sorting : Insertion sort, Heap sort, Bubble sort

6. Searching : Sequential and Binary Search

7. Selection : Minimum/ Maximum, Kth smallest element.
Objective
Computer Graphics is the illustration field of Computer Science. This subject focuses on the design, modeling, analysis, and applications of computer-related systems.

Theory
Note: Total five questions are to be attempted from Sections A & B.

Section-A

Unit-1

Unit-2

Unit-3
Two Dimensional Transformation: What is transformation? Matrix representation of points, Basic transformation: translation, scaling, rotation, Rotation about an arbitrary point. Reflection, composite transformation.

Three-dimensional transformations: Three dimensional graphics concept, Matrix representation of 3-D Transformations, Composition of 3-D transformation.

Section-B

Unit-4
Two/Three Dimensional Viewing: The 2-D viewing pipeline, windows, viewports, window to view port mapping;

Clipping: point, clipping line (algorithms):- 4 bit code algorithm, Sutherland-cohen algorithm, parametric line clipping algorithm (Cyrus Beck).

Polygon clipping algorithm: Sutherland-Hodgeman polygon clipping algorithm.

Unit-5
Viewing in 3D: Projections, types of projections, the mathematics of planner geometric projections,

Unit-6

Representing Curves and Surfaces: Parametric representation of curves: Bezier curves, B-Spline curves. Parametric representation of surfaces;

Illumination: Illumination models, Shading, shading models for polygons, shadows, transparency.

Text Books :-
1. Computer graphics, Hearn and Baker, PHI
2. Computer Graphics, Foley, PE-LPE,

Reference Books
1. Procedural Elements of Computer graphics, Rogers, McGraw Hill

CG Lab Experiments

List of programs to be developed:

1. Introduction to Graphics and graphic functions.

2. Write a program to draw a line using DDA Algorithm.

3. Write a program to draw a line using Bresenham’s Algorithm.

4. Write a program to draw a circle using Bresenham’s Algorithm.

5. Write a program to translate a triangle.

6. Write a program to scale a triangle.

7. Write a program to rotate a triangle with respect to fixed point.

8. Write a program to scale a triangle with respect to fixed point.

9. Write a program for line clipping.

10. Write a program for Window to viewport mapping.
Objective: This course provides an introduction to the fundamentals of distributed computer systems, assuming the availability of facilities for data transmission. The structure of distributed systems using multiple levels of software is emphasized.

Theory:
Note: Total five questions are to be attempted from Sections A & B.

Section A

Unit-1:

Unit-2:
Synchronization in Distributed System: Clock synchronization, Mutual Exclusion, Election algorithm, the Bully algorithm, a Ring algorithm, Atomic Transactions.

Unit-3:
Deadlock in Distributed Systems, Distributed Deadlock Prevention, Distributed Deadlock Detection.

Section B

Unit-4:

Unit-5:
Distributed file systems: Distributed file system Design, Distributed file system Implementation, Trends in Distributed file systems.

Unit-6:
Distributed Shared Memory: What is shared memory, Consistency models, Page based distributed shared memory, shared variables distributed shared memory.
Text Book:

1 Distributed Operating System – Andrew S. Tanenbaum, PHI.

2 Operating System Concepts , P.S.Gill, Firewall Media
Objective
This subject focuses on the brief details of artificial intelligence concepts & all techniques used to implement this.

Theory
Note: Total five questions are to be attempted from Sections A & B.

Section-A

Unit 1: Foundational issues in intelligent systems: Foundation and history of AI, AI problems and techniques AI programming languages, introduction to LISP and PROLOG- problem spaces and searches.

Unit 2: Searching Techniques

Unit 3: Knowledge Representation
Knowledge representation issues, predicate logic- logic programming, semantic nets- frames, constraint propagation, representing knowledge using rules

Section-B

Unit 4: Reasoning
Reasoning under uncertainty, review of probability, Baye” s probabilistic interferences, Symbolic reasoning under uncertainty, Statistical reasoning, Fuzzy reasoning, Temporal reasoning, Non monotonic reasoning.

Unit 5: Planning & Learning
Planning, planning in situational calculus, representation for planning, partial order planning algorithm, learning from examples, discovery as learning, Learning by analogy, explanation based learning.

Unit 6: Expert System
Principles of Natural language processing, rule based systems architecture, Expert systems, knowledge acquisition concepts, AI application to robotics, neural nets, and genetic algorithms
Text Book:


Reference Books:

1) Artificial Intelligence, Elain Rich and Kevin Knight, 1991, TMH.
2) Artificial Intelligence-A modern approach, Staurt Russel and peter norvig, 1998, PHI.
3) Artificial intelligence, by Janaki & Raman
4) Artificial Intelligence by Patterson

Artificial Intelligence Lab:

1. Study of PROLOG.
Write the following programs using PROLOG.
1. Write a program to find factorial of any number.
2. Write a program to find maximum of 3 numbers
3. Write a program to solve tower of Hanoi.
4. Solve any problem using depth first search.
5. Solve any problem using best first search.
6 Solve 8-puzzle problem using best first search.
7. Solve traveling salesman problem.
8. Write a program to find largest, smallest element from a list.
9. Write a program to find sum of two numbers.
10. Implement various operations on a list like:- deletion in a list, insertion in a list, creation of a list, palindrome or not, member or not.
### Objective:

The goal of the course is to provide an introduction to the system software like assemblers, compilers, and macros. It provides the complete description about inner working of a compiler. The main focus is on the design of compilers and optimization techniques. The course also aims to convey the language specifications, use of regular expressions and context free grammars behind the design of compiler.

The objective of this course is to provide a student with an understanding of the fundamental principles in compiler design and to provide the skills needed for building compilers for various situations that one may encounter in a career in Computer Science.

### Theory

**Note:** Total five questions are to be attempted from Sections A & B.

#### SECTION–A

**Unit–1: Introduction To Compilers:** Compilers and translators, need of translators, structure of compiler: its different phases, Compiler construction tools.

**Unit–2: Lexical Analysis:** Role of lexical analyzer, design of lexical analyzer, regular expressions, Specification and recognition of tokens, input buffering, A language specifying lexical analyzer. Finite automata, conversion from regular expression to finite automata, and vice versa, minimizing number of states of DFA, Implementation of lexical analyzer.

**Unit–3: Syntax Analysis:** Role of parsers, context free grammars, definition of parsing.

**Parsing Technique:** Shift- reduce parsing, operator precedence parsing, top down parsing, predictive parsing.

#### SECTION – B

**Unit–4:** LR parsers, SLR, LALR and Canonical LR parser.

**Unit–5: Syntax Directed Translations:** Syntax directed definition, construction of syntax trees, syntax directed translation scheme, and implementation of syntax directed translation, three address code, quadruples and triples.

**Unit–6: Symbol Table & Error Detection And Recovery:** Symbol tables, its contents and data structure for symbol tables; trees, arrays, linked lists, hash tables. Errors, lexical phase error, syntactic phase error, semantic error.
Text Books:


Reference Books:

2. System software by Dhamdae, 1986, MGH.

Principle of compiler Design Lab:

1. To show all the operations of a stack.
2. Write a program to check whether a string belong to the grammar or not.
3. Write a program to generate a parse tree.
4. Write a program to find leading terminals.
5. Write a program to find trailing terminals.
6. Write a program to compute FIRST of non-terminal.
7. Write a program to compute FOLLOW of non-terminal.
8. Write a program to check whether a grammar is left Recursion and remove left Recursion.
9. Write a program to remove left factoring.
10. Write a program to check whether a grammar is operator precedent.
Objective:

This course fully covers computer networks fundamentals and different types of networks.

NOTE: Question No 1 is compulsory and will be of short answer type from entire syllabus. Two questions are to be attempted out of three questions from each Section A & B.

Section-A

Unit-1 Introduction:
Data communications: components, data representation (ASCII, ISO etc.), direction of data flow (simplex, half duplex, full duplex), network criteria, Introduction to Computer Networks, Network Topologies: Bus-, Star-, Ring-, Hybrid -, Tree -, Complete -, Irregular – Topology; Types of Networks : Local Area Networks, Metropolitan Area Networks, Wide Area Networks;

Unit-2 OSI & TCP Reference Model:
Layering architecture of networks, OSI model, Functions of each layer, Services and Protocols of each layer, History of TCP/IP, Layers of TCP/IP, Protocols,

Unit-3 TCP:

Section-B

Unit-4 Local Area Networks:
Introduction to LANs, Features of LANs, Components of LANs, Usage of LANs, LAN Standards, Channel Access Methods, Aloha, CSMA, CSMA/CD, Token Passing, Ethernet, Fast Ethernet and Gigabit Ethernet, Token Ring, LAN interconnecting devices: Hubs, Switches, Bridges, Routers, Gateways.

Metropolitan Area Network: Distributed queue dual bus (DQDB).

Unit-5 Wide Area Networks:
Introduction of WANs, Routing, Congestion Control, WAN Technologies, Synchronous Digital Hierarchy (SDH)/ Synchronous Optical Network (SONET), Asynchronous Transfer Mode (ATM), Frame Relay.

Text Book:

Reference Books:

LAB:

Note: At least ten experiments are to be performed during the semester. Five experiments may be designed by the concern faculty in consultation with H.O.D as per the scope of syllabus.

Objective: To understand the concepts of computer networks.

List of Experiments:

1. Study of different types of Network cables. Practically implement the cross-wired cable using climping tool and use this cross wire to connect the systems for sharing data.
2. Practically implement the cross-wired cable using climping tool .Connect the computers in Local Area Network in star topology.
3. Study of Network Devices in Detail.
4. Study of network IP
   a. Classification of IP address
   b. Sub netting
   c. Super netting
5. Study of basic network command and Network configuration commands.
6. Write a program to find IP address and host name of host machine.
7. Write a program to implement day time TCP client using JAVA/C.
8. Write a program to implement TCP echo client server using JAVA/C.
9. Write a program to implement UDP echo client server using JAVA/C.
10. Write a program to implement TCP multithreaded server handling more than one client using JAVA/C.
11. Configure a Network topology using packet tracer software.
12. Configure a Network using Distance Vector Routing protocol: RIP.
13. Configure Network using Link State Vector Routing protocol: OSPF.
Objective
This subject focuses better understanding and deeper knowledge of the cyber security. It also focuses on various concepts like computer security, Hacking Techniques, Forensics etc.

Theory
Note: Total five questions are to be attempted from Sections A & B.

SECTION – A

Unit 1: Networks and the Internet
Introduction to Network Basics, General Architecture of Internet, IP Addresses, Uniform Resource Locators and their role, Basic Network Utilities, IPConfig, Ping, Tracert.

Unit 2: Introduction to Computer Security
Importance of Information and its Security, Types of Threats: Malware, Password attack, Denial of Service Attacks, Web Attacks, DNS Poisoning, Packet Spoofing, Session Hijacking; Basic Security Terminology.

Unit 3: Hacking Techniques

SECTION – B

Unit 4: Cyber Attacks
Introduction to Internet Frauds, Socially engineered Trojans, Phishing, Cyber Stalking, Types of Attacks: Spoofing, Man in the Middle, Identity Theft, Protecting Yourself against Cyber Crime, Protecting against Identity Theft, Secure Browser Settings.

Unit 5: Introduction to Forensics

Unit 6: Cyber Rules and Regulation

Reference Books:
2. Laws on Cyber Crimes-Dr. Pramod Kr. Singh, Book Enclave, Jaipur.
Note: Eight questions will be set in all by the examiners taking at least one question from each unit. Students will be required to attempt five questions in all.

**CYBER SECURITY LAB:**

Note: At least ten experiments are to be performed during the semester. At least eight experiments should be performed from the list of experiments. Two experiments may either be performed from the given list of experiments or may be designed by the concerned faculty in consultation with H.O.D as per the scope of syllabus.

**Objective:** To understand the concepts of security.

**List of Experiments:**

1. To protect office document from unauthorized person.
2. To provide user to access a file in LINUX
3. To configure IP address and study of various network commands IPConfig, Ping, Tracert.
4. Applying Audit policy.
5. Setting up the local security policy.
6. Setting up the firewalls.
7. Finding evidence in the browser.
8. Implement Subnet Masking in brief.
9. Management of the IIS and FTP server
10. Start and stop services from user window and command prompt.
11. Implementation of cyber attacks.
12. Implementation of session Hijacking.
**Objective:** Java is a computer programming language that is concurrent, class-based, object-oriented, and specifically designed to have as few implementation dependencies as possible.

**Theory:**

**Note:** Question No 1 is compulsory and will be of short answer type from entire syllabus. Two questions are to be attempted out of three questions from each Section A & B.

**Section A**

**Unit-1:**
Core Java:
Introduction to Java, Data types, variables, operators, Arrays, Control Statements, Classes & Methods, Packages, Inheritance, Utility classes, Multithreading, Collections, I/O streams, AWT & Applet Programming.

**Unit-2:**
Event Handling and Networking:

**Unit-3:**

**Section B**

**Unit-4:**
Servlets
Servlet Overview and Architecture, Interface Servlet and the Servlet Life Cycle, Handling HTTP get Requests, Handling HTTP post Requests, Redirecting Requests to Other Resources, Session Tracking, Cookies, Session Tracking with HttpSession.

**Unit-5:**
JavaServer Pages (JSP)
Unit-6:

EJB and RMI

**Enterprise Java Bean:** Preparing a Class to be a JavaBean, Creating a JavaBean, JavaBean Properties, Types of beans, Stateful Session bean, Stateless Session bean, Entity bean.

**Remote Method Invocation:** Defining the Remote Interface, Implementing the Remote Interface, Compiling and Executing the Server and the Client

**TEXT:**


**REFERENCE BOOK:**

1. Professional Java Programming by Brett Spell, WROX Publication
3. Advanced Java, Gajendra Gupta, Firewall Media.

**Advance Java Lab:**

List of programs

1) Write a program to implement multiple inheritance in java
2) Write a program to show the concept of synchronization in Multithreading.
3) Write a program to pass parameters in Applet.
4) Write a program to execute select query using JDBC
5) Write a program to Update Customer Information using JDBC-ODBC driver.
6) Write a program of simple servlet that just generates plain text
7) Write a program to display a String using JSP.
8) Write a program to create a calculator (performing addition, subtraction, multiplication and division) using Remote Method Invocation(RMI).
9) Write a program to add two numbers given by the user in two textboxes and show their sum in the third textbox using applets in java.
10) Write a program to create check boxes using JSP.
Any technical subject can be taught which is suitable according to current industrial need and which is not being taught as part of current curriculum, decided by a departmental committee which headed by the head of the department.