# PONDICHERRY ENGINEERING COLLEGE, PUDUCHERRY – 605 014

## CURRICULUM AND SYLLABI FOR AUTONOMOUS STREAM

### MASTER OF COMPUTER APPLICATIONS (M.C.A) COURSE

(For students admitted from academic year 2015-16 onwards)

## CURRICULUM

### I SEMESTER

<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Category</th>
<th>Periods</th>
<th>Marks</th>
<th>Credit</th>
</tr>
</thead>
<tbody>
<tr>
<td>MA153</td>
<td>Mathematical Foundation of Computer Science</td>
<td>TY</td>
<td>3 1 -</td>
<td>40 60 100</td>
<td>4</td>
</tr>
<tr>
<td>CA151</td>
<td>Digital System Design</td>
<td>TY</td>
<td>3 1 -</td>
<td>40 60 100</td>
<td>4</td>
</tr>
<tr>
<td>CA152</td>
<td>Data Structures</td>
<td>TY</td>
<td>3 1 -</td>
<td>40 60 100</td>
<td>4</td>
</tr>
<tr>
<td>CA153</td>
<td>Problem Solving and Programming</td>
<td>TY</td>
<td>3 1 -</td>
<td>40 60 100</td>
<td>4</td>
</tr>
<tr>
<td>CA154</td>
<td>Operating Systems</td>
<td>TY</td>
<td>3 1 -</td>
<td>40 60 100</td>
<td>4</td>
</tr>
<tr>
<td>CA155</td>
<td>C and Data Structures Laboratory</td>
<td>LB</td>
<td>- - 3</td>
<td>60 40 100</td>
<td>2</td>
</tr>
<tr>
<td>CA156</td>
<td>Operating Systems Laboratory</td>
<td>LB</td>
<td>- - 3</td>
<td>60 40 100</td>
<td>2</td>
</tr>
</tbody>
</table>

**Total Credits**: 24

### II SEMESTER

<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Category</th>
<th>Periods</th>
<th>Marks</th>
<th>Credit</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA157</td>
<td>Design and Analysis of Algorithms</td>
<td>TY</td>
<td>3 1 -</td>
<td>40 60 100</td>
<td>4</td>
</tr>
<tr>
<td>CA158</td>
<td>Object Oriented Programming</td>
<td>TY</td>
<td>3 1 -</td>
<td>40 60 100</td>
<td>4</td>
</tr>
<tr>
<td>CA159</td>
<td>Computer Organization and Architecture</td>
<td>TY</td>
<td>3 1 -</td>
<td>40 60 100</td>
<td>4</td>
</tr>
<tr>
<td>CA160</td>
<td>Business Process</td>
<td>TY</td>
<td>3 1 -</td>
<td>40 60 100</td>
<td>4</td>
</tr>
<tr>
<td>CA161</td>
<td>Computer Graphics and Animation</td>
<td>TY</td>
<td>3 1 -</td>
<td>40 60 100</td>
<td>4</td>
</tr>
<tr>
<td>CA162</td>
<td>Algorithms Laboratory</td>
<td>LB</td>
<td>- - 3</td>
<td>60 40 100</td>
<td>2</td>
</tr>
<tr>
<td>CA163</td>
<td>Object Oriented Programming Laboratory</td>
<td>LB</td>
<td>- - 3</td>
<td>60 40 100</td>
<td>2</td>
</tr>
</tbody>
</table>

**Total Credits**: 24
### III SEMESTER

<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject Category</th>
<th>Periods</th>
<th>Marks</th>
<th>Credit</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA164</td>
<td>Database Management Systems</td>
<td>TY</td>
<td>3 1 -</td>
<td>40 60 100</td>
</tr>
<tr>
<td>CA165</td>
<td>Computer Networks</td>
<td>TY</td>
<td>3 1 -</td>
<td>40 60 100</td>
</tr>
<tr>
<td>CA166</td>
<td>Platform Technologies</td>
<td>TCM</td>
<td>3 - 2</td>
<td>50 50 100</td>
</tr>
<tr>
<td>-</td>
<td>Elective – I</td>
<td>TY</td>
<td>3 - -</td>
<td>40 60 100</td>
</tr>
<tr>
<td>-</td>
<td>Elective - II</td>
<td>TY</td>
<td>3 - -</td>
<td>40 60 100</td>
</tr>
<tr>
<td>CA167</td>
<td>Database Management Systems Laboratory</td>
<td>LB</td>
<td>- - 3</td>
<td>60 40 100</td>
</tr>
<tr>
<td>CA168</td>
<td>Computer Networks Laboratory</td>
<td>LB</td>
<td>- - 3</td>
<td>60 40 100</td>
</tr>
<tr>
<td>HS151</td>
<td>Communication Skills</td>
<td>PR</td>
<td>- - 3</td>
<td>100 - 100</td>
</tr>
</tbody>
</table>

Total Credits: **23**

### IV SEMESTER

<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject Category</th>
<th>Periods</th>
<th>Marks</th>
<th>Credit</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA169</td>
<td>Web Technologies</td>
<td>TY</td>
<td>3 1 -</td>
<td>40 60 100</td>
</tr>
<tr>
<td>CA170</td>
<td>Theory of Computation</td>
<td>TY</td>
<td>3 1 -</td>
<td>40 60 100</td>
</tr>
<tr>
<td>CA171</td>
<td>Software Engineering</td>
<td>TY</td>
<td>3 1 -</td>
<td>40 60 100</td>
</tr>
<tr>
<td>-</td>
<td>Elective – III</td>
<td>TY</td>
<td>3 - -</td>
<td>40 60 100</td>
</tr>
<tr>
<td>-</td>
<td>Elective - IV</td>
<td>TY</td>
<td>3 - -</td>
<td>40 60 100</td>
</tr>
<tr>
<td>CA172</td>
<td>Web Technologies Laboratory</td>
<td>LB</td>
<td>- - 3</td>
<td>60 40 100</td>
</tr>
<tr>
<td>CA173</td>
<td>CASE Tools Laboratory</td>
<td>LB</td>
<td>- - 3</td>
<td>60 40 100</td>
</tr>
</tbody>
</table>

Total Credits: **22**
## V SEMESTER

<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Name of the Subject</th>
<th>Category</th>
<th>Periods</th>
<th>Marks</th>
<th>Credit</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA174</td>
<td>Mobile Computing</td>
<td>TY</td>
<td>3 1 -</td>
<td>40 60 100</td>
<td>4</td>
</tr>
<tr>
<td>CA175</td>
<td>Management Concepts and Strategies</td>
<td>TY</td>
<td>3 1 -</td>
<td>40 60 100</td>
<td>4</td>
</tr>
<tr>
<td>-</td>
<td>Elective - V</td>
<td>TY</td>
<td>3 - -</td>
<td>40 60 100</td>
<td>3</td>
</tr>
<tr>
<td>-</td>
<td>Elective – VI</td>
<td>TY</td>
<td>3 - -</td>
<td>40 60 100</td>
<td>3</td>
</tr>
<tr>
<td>-</td>
<td>Elective - VII</td>
<td>TY</td>
<td>3 - -</td>
<td>40 60 100</td>
<td>3</td>
</tr>
<tr>
<td>CA176</td>
<td>Mobile Application Development Laboratory</td>
<td>LB</td>
<td>- - 3</td>
<td>60 40 100</td>
<td>2</td>
</tr>
<tr>
<td>CA177</td>
<td>Mini Project</td>
<td>PR</td>
<td>- - 3</td>
<td>60 40 100</td>
<td>2</td>
</tr>
<tr>
<td>CA178</td>
<td>Comprehensive Test and Viva - Voce</td>
<td>PR</td>
<td>- - 3</td>
<td>60 40 100</td>
<td>1</td>
</tr>
</tbody>
</table>

Total Credits 22

## VI SEMESTER

<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Name of the Subject</th>
<th>Category</th>
<th>Periods</th>
<th>Marks</th>
<th>Credit</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA179</td>
<td>Project Work</td>
<td>PR</td>
<td>- - -</td>
<td>150 150 300</td>
<td>10</td>
</tr>
<tr>
<td>-</td>
<td>Professional Development Courses(Two one credit courses)</td>
<td>PR</td>
<td>- - -</td>
<td>200 - 200</td>
<td>2</td>
</tr>
</tbody>
</table>

Total Credits 12

A representative list of *Professional Development Courses* is given below (Limited to one credit):

a) Industrial Training
b) Specific Field Knowledge Training
c) Seminar related with directed study
d) Foreign Language Learning Certificate - offered by the college or by a recognized agency duly approved by the college

# CA – Continuous Assessment, SE – Semester Examination, TM – Total Marks
* TY – Theory, LB – Laboratory, PR–Practice, TCM – Theory combined with Mini Project
### LIST OF ELECTIVES

<table>
<thead>
<tr>
<th>Sl.No.</th>
<th>Subject Code</th>
<th>Subject</th>
<th>Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CAE51</td>
<td>Object Oriented Analysis and Design</td>
<td>TY</td>
</tr>
<tr>
<td>2</td>
<td>CAE52</td>
<td>System Software</td>
<td>TY</td>
</tr>
<tr>
<td>3</td>
<td>CAE53</td>
<td>Information Security</td>
<td>TY</td>
</tr>
<tr>
<td>4</td>
<td>CAE54</td>
<td>Cloud Computing</td>
<td>TY</td>
</tr>
<tr>
<td>5</td>
<td>CAE55</td>
<td>Hardware and Troubleshooting</td>
<td>TY</td>
</tr>
<tr>
<td>6</td>
<td>CAE56</td>
<td>Accounting and Financial Management</td>
<td>TY</td>
</tr>
<tr>
<td>7</td>
<td>CAE57</td>
<td>Resource Management Techniques</td>
<td>TY</td>
</tr>
<tr>
<td>8</td>
<td>CAE58</td>
<td>Agent Technologies</td>
<td>TY</td>
</tr>
<tr>
<td>9</td>
<td>CAE59</td>
<td>UNIX Internals</td>
<td>TY</td>
</tr>
<tr>
<td>10</td>
<td>CAE60</td>
<td>Software Architecture</td>
<td>TY</td>
</tr>
<tr>
<td>11</td>
<td>CAE61</td>
<td>Social Network Analysis</td>
<td>TY</td>
</tr>
<tr>
<td>12</td>
<td>CAE62</td>
<td>Advanced Java Programming</td>
<td>TY</td>
</tr>
<tr>
<td>13</td>
<td>CAE63</td>
<td>Data Mining and Warehousing</td>
<td>TY</td>
</tr>
<tr>
<td>14</td>
<td>CAE64</td>
<td>Artificial Intelligence</td>
<td>TY</td>
</tr>
<tr>
<td>15</td>
<td>CAE65</td>
<td>Principles of Distributed Systems</td>
<td>TY</td>
</tr>
<tr>
<td>16</td>
<td>CAE66</td>
<td>Distributed Database System</td>
<td>TY</td>
</tr>
<tr>
<td>17</td>
<td>CAE67</td>
<td>Software Testing and Quality Assurance</td>
<td>TY</td>
</tr>
<tr>
<td>18</td>
<td>CAE68</td>
<td>Software Project Management</td>
<td>TY</td>
</tr>
<tr>
<td>19</td>
<td>CAE69</td>
<td>Parallel Computing</td>
<td>TY</td>
</tr>
<tr>
<td>20</td>
<td>CAE70</td>
<td>Big Data Analytics</td>
<td>TY</td>
</tr>
<tr>
<td>21</td>
<td>CAE71</td>
<td>User Interface Design</td>
<td>TY</td>
</tr>
<tr>
<td>22</td>
<td>CAE72</td>
<td>Multimedia Systems and Applications</td>
<td>TY</td>
</tr>
<tr>
<td>23</td>
<td>CAE73</td>
<td>Microprocessors and Assembly Language Programming</td>
<td>TY</td>
</tr>
</tbody>
</table>
SYLLABUS (Core Subjects)
### Department: Mathematics

**Programme:** Master of Computer Application

**Semester:** One  
**Category:** TY

<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>MA153</td>
<td>Mathematical Foundation of Computer Science</td>
<td>3 L, 1 T, - P</td>
<td>4 C</td>
<td>40 CA, 60 SE, 100 TM</td>
</tr>
</tbody>
</table>

**Prerequisite:**
To familiarize the students with
- Rules and Techniques to recognize valid logical argument.
- The basic idea of logic with the algebra of proposition and predicate logic.
- The idea of sets, functions and basic Graph theory.

### Objectives
On successful completion of this course, students will be able to
- Develop knowledge of logical connectivity, compound propositions, formal.
- Symbols of propositional logic and find exact value of expressions.
- Use the formal symbol to predicate logic.
- Apply graph theory in real time network problems, data structures etc.

### Outcomes

#### UNIT – I Mathematical Logic


#### UNIT – II Normal Forms and Inference Theory


#### UNIT – III Predicate Calculus

- Predicates - The statement function, variables and quantifiers - Predicate formulas - symbolizing the statement - Inference theory of the predicate calculus - Rules of specification and generalization - Derivation of conclusion using the rules of inference theory.

#### UNIT – IV Set Theory


#### UNIT – V Graph Theory


**Total Contact Hours:** 48  
**Total Tutorials:** 12  
**Total Practical Classes:** -  
**Total Hours:** 60

### Text Books:


### Reference Books:


### Websites:
-
**Department:** Computer Science and Engineering  
**Programme:** Master of Computer Application  
**Semester:** One  
**Category:** TY  

<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA151</td>
<td>Digital System Design</td>
<td>L 3 T 1 P - C 4</td>
<td>CA 40 SE 60 TM 100</td>
<td></td>
</tr>
</tbody>
</table>

**Prerequisite:** -  

**Objectives**  
- To introduce the fundamentals of digital system design.  
- To lay strong foundation in combinational and sequential logic fundamentals.  
- To educate from basic concepts to advanced system design.  

**Outcomes**  
On successful completion of the course, students will be able to:  
- Understand the binary number systems and Boolean algebra.  
- Design any combinational logic using only of universal gates, MSI gates and PLDs  
- Design and implement sequential logic circuits of any complexity.  
- Simulate and validate the correctness of the digital circuits using VHDL packages.  

**UNIT – I Number Systems and Boolean Algebra**  
Hours: 09  
Revision of RTL, DTL, I L, TTL, ECL, MOS, CMOS logic families - Binary number systems and conversion - Binary arithmetic-Binary codes - Boolean algebra - Basic operations - Basic Theorems - Boolean functions-Canonical forms - Simplification of Boolean functions-Karnaugh maps - Tabulation method.  

**UNIT – II Combinational Logic**  
Hours: 09  

**UNIT – III Sequential Circuits**  
Hours: 09  
Registers and Counters: Registers – shift registers – ripple counters – synchronous counters – other counters  

**UNIT – IV Memory and Programmable Logic**  
Hours: 09  

**UNIT – V Introduction to VHDL**  
Hours: 09  
introduction – VHDL Design flow, program structure, types and constants, functions and procedures, libraries and packages – VHDL for combinational circuits, Sequential Circuits, Registers and Counters, VHDL description for binary multiplier.  

**Total contact Hours:** 45  
**Total Tutorials:** 15  
**Total Practical Classes:** -  
**Total Hours:** 60  

**Text Books:**  

**Reference Books:**  

**Websites:**  
1. [http://www.nptel.iitm.ac.in/video.php?subjectId=117106086](http://www.nptel.iitm.ac.in/video.php?subjectId=117106086)  
2. [http://www.xilinx.com](http://www.xilinx.com)  
## Data Structures

**Course Code:** CA152  
**Subject:** Data Structures

<table>
<thead>
<tr>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>L T P C CA SE TM</td>
<td>3 1 -</td>
<td>40 60 100</td>
</tr>
</tbody>
</table>

### Prerequisite

- None

### Objectives

- To gain knowledge about storage and manipulation of data used for programming.
- To emphasize the concept of data abstraction and the problem of building implementations of abstract data types.

### Outcomes

On successful completion of the module students will be able to:

- Select relevant data structures and combinations of relevant data structures for the given problems in terms of memory and run time efficiency.
- Apply data abstraction in solving programming problems.

## Content

### UNIT – I  Arrays and Searching Algorithms  Hours: 09


### UNIT – II  Linear Data Structures  Hours: 09


### UNIT – III  Non-Linear Data Structures  Hours: 09


### UNIT – IV  Hash Tables and Search Trees  Hours: 09

Hash Table: Hash Functions, Collision Resolution Strategies, Hash Table, Implementation. Binary Search Trees: Binary Search Tree (BST), Insertion and Deletion in BST, Complexity of Search Algorithm, Path Length, AVL Trees, B-trees.

### UNIT – V  File Structures  Hours: 09

Physical Storage Media File Organization, Organization of records into Blocks, Sequential Files- Read/Write Operations, Indexing, Primary indices, Secondary indices, B+ Tree index Files, B Tree index Files, Comparison of Hashing and Indexing techniques.

### Total

- Total contact Hours: 45  
- Total Tutorials: 15  
- Total Practical Classes: -  
- Total Hours: 60

### Text Books:


### Reference Books:


### Websites:

1. http://www.cse.unt.edu  
2. http://nptel.iitm.ac.in
<table>
<thead>
<tr>
<th>Department</th>
<th>Programme</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computer Science and Engineering</td>
<td>Master of Computer Application</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Semester</th>
<th>Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>One</td>
<td>TY</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA153</td>
<td>Problem Solving and Programming</td>
<td>3 1 - 4</td>
<td>4</td>
<td>100</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Prerequisite</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Objectives</th>
</tr>
</thead>
<tbody>
<tr>
<td>• To introduce basic understanding of design, and implementation of computing,</td>
</tr>
<tr>
<td>programming and problem-solving.</td>
</tr>
<tr>
<td>• To develop basic programming skills (logic, ability) to solve problems &amp;</td>
</tr>
<tr>
<td>practice of program writing</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Outcome</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Selection of appropriate language constructs, design and implementation of problem solving</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>UNIT – I</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introduction to Problem Solving</td>
</tr>
<tr>
<td>Hours: 09</td>
</tr>
<tr>
<td>Problem solving strategies, Problem identification, Problem understanding,</td>
</tr>
<tr>
<td>Algorithm development, Solution planning (flowcharts, pseudo - code, etc.),</td>
</tr>
<tr>
<td>Modular programming design. Basic program structure in C, Simple data</td>
</tr>
<tr>
<td>types, variables, constants, operators, comments, Control Flow; if, while,</td>
</tr>
<tr>
<td>for, do - while, switch.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>UNIT – II</th>
</tr>
</thead>
<tbody>
<tr>
<td>Functions, Arrays and Pointers</td>
</tr>
<tr>
<td>Hours: 09</td>
</tr>
<tr>
<td>Types, parameters, prototypes, recursion. Arrays &amp; Pointers: Array usage,</td>
</tr>
<tr>
<td>Strings – arrays of pointers. Arguments to main. Pointers to functions.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>UNIT – III</th>
</tr>
</thead>
<tbody>
<tr>
<td>Structures and Linked Lists</td>
</tr>
<tr>
<td>Hours: 09</td>
</tr>
<tr>
<td>Member accessing, pointers to structures, Structures and functions,</td>
</tr>
<tr>
<td>Arrays of structures, linked lists, trees. Other Data Types: Unions,</td>
</tr>
<tr>
<td>enumerations and bit fields</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>UNIT – IV</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bitwise Operators, Dynamic Allocation and Pre-processors</td>
</tr>
<tr>
<td>Hours: 09</td>
</tr>
<tr>
<td>Bitwise Operators: Usage, device accessing. Type manipulation: Coercion,</td>
</tr>
<tr>
<td>typedef, initialization, Static, global, external, register. Dynamic</td>
</tr>
<tr>
<td>Allocation: Uses, pitfalls. The Pre-processor directives- macro</td>
</tr>
<tr>
<td>definitions and usage</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>UNIT – V</th>
</tr>
</thead>
<tbody>
<tr>
<td>File Operations</td>
</tr>
<tr>
<td>Hours: 09</td>
</tr>
<tr>
<td>Input and Output: Concepts, Character and File I/O, Simple File I/O, File</td>
</tr>
</tbody>
</table>

| Total contact Hours: 45 | Total Tutorials: 15 | Total Practical Classes: - | Total Hours: 60 |

<table>
<thead>
<tr>
<th>Text Books:</th>
</tr>
</thead>
<tbody>
<tr>
<td>India, 2005.</td>
</tr>
<tr>
<td>3. Dromey, How To Solve It By Computer, Dorling Kindersley PvtLtd,Second</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Reference Books:</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. E.Balagurusamy, Computing fundamentals and C Programming, Tata McGraw-</td>
</tr>
<tr>
<td>2. BehrouzA.Forouzan and Richard F. Gilberg, A Structured Programming</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Websites:</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. <a href="http://www.cse.unt.edu">http://www.cse.unt.edu</a></td>
</tr>
<tr>
<td>2. <a href="http://nptel.iitm.ac.in">http://nptel.iitm.ac.in</a></td>
</tr>
</tbody>
</table>
**CA154 Operating Systems**

**Subject Code**
CA154

**Subject**
Operating Systems

**Hours / Week**
L: 3, T: 1, P: 1, C: 4

**Credit**
4

**Maximum Marks**
40

**Total Marks**
100

**Prerequisite**
- None

**Objectives**
- To be aware of the evolution and fundamental principles of operating system, processes and their communication
- To understand the various operating system components like process management, memory management and file management

**Outcomes**
On successful completion of the module students will be able to:
- Understand the operating system components and its services.
- Realize the intricacies of Process Synchronization.
- Demonstrate the mapping between the physical memory and virtual memory.

**UNIT – I Introduction**

- Types of operating systems
- Operating systems structures
- Systems components
- Operating systems services
- System calls
- Systems programs
- Processes
- Process concept
- Process scheduling
- CPU scheduling
- Scheduling criteria
- Scheduling algorithms
- Multiple-processor Scheduling

**UNIT – II Process Synchronization**

- Critical Section problem
- Semaphores
- Classical problems of synchronization
- Critical regions
- Monitors
- Deadlock
- Deadlock handling
- Deadlock Prevention
- Deadlock avoidance
- Deadlock Detection
- Deadlock Recovery
- Threads
- Multithreading Models

**UNIT – III Memory Management**

- Swapping
- Contiguous Memory allocation
- Paging
- Segmentation
- Virtual Memory
- Demand paging
- Page Replacement
- Thrashing

**UNIT – IV Disk Management**

- Disk Structures
- Disk Scheduling
- File Systems Interface
- File concepts
- Access methods
- Directory Structures
- Directory Implementation
- Allocation Methods
- Free Space management

**UNIT – V Case Studies**

- Linux System design Principles
- process management
- File Systems
- Windows Vista
- Systems Structures
- Process management
- memory management
- Android OS
- Virtual machine OS

**Total contact Hours:** 45
**Total Tutorials:** 15
**Total Practical Classes:** -
**Total Hours:** 60

**Text Books:**

**Reference Books:**

**Websites:**
**Department**: Computer Science and Engineering  
**Programme**: Master of Computer Application  
**Semester**: One  
**Category**: LB

<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA155</td>
<td>C and Data Structures Laboratory</td>
<td>-</td>
<td>3</td>
<td>60</td>
</tr>
</tbody>
</table>

**Prerequisite**: -

**Objectives**
- To develop skills to design and analyze simple linear and non linear data structures
- To Strengthen the ability to identify and apply the suitable data structure for the given real world problem
- To Gain knowledge in practical applications of data structures

**Outcomes**
On successful completion of the course, students will be able to:
- To design and analyze the time and space efficiency of the data structure
- To identity the appropriate data structure for given problem
- Employ a deep knowledge of various data structures when constructing a program.

**Cycle - I**  
**Hours: 12**
1. C Programs – Arrays, Strings, Structures, Functions and Files  
2. Searching algorithms - sequential, binary and Fibonacci search algorithms on an ordered list. Compare the number of key comparisons made during the searches  
3. Sorting algorithms : Insertion Sort, Selection Sort, Bubble Sort and Quick Sort

**Cycle - II**  
**Hours: 33**
1. Sparse matrix representation and find its transpose.  
2. Evaluation of arithmetic expression to postfix expression.  
3. Queue, circular queue  
4. Singly Linked List, Doubly Linked List, Circular Linked List  
5. Concatenation of linked lists.  
6. Tree traversals - insertions and deletions  
7. Graph traversals  
8. Implementation of Hash tables

**Total contact Hours**: -  
**Total Tutorials**: -  
**Total Practical Classes**: 45  
**Total Hours**: 45
<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA156</td>
<td>Operating Systems Laboratory</td>
<td>-</td>
<td>3</td>
<td>60</td>
</tr>
</tbody>
</table>

**Prerequisite**

- Learn shell programming in the UNIX environment.
- Be exposed to programming in C using system calls.
- Be exposed to process creation and inter process communication.
- Be familiar with implementation of CPU Scheduling Algorithms, page replacement algorithms and Deadlock avoidance.

**Objectives**

- On successful completion of the course, students will be able to:
  - Implement deadlock avoidance.
  - Compare the performance of various CPU Scheduling Algorithms.
  - Critically analyze the performance of the various page replacement algorithms.
  - Create processes and implement IPC with process synchronization.

**Cycle - I**

### Fundamentals

1. Learn the use of basic UNIX commands.
2. Shell Programming.
3. Implement the following CPU scheduling algorithms a) Round Robin b) SJF c) FCFS d) Priority
4. Implement all file allocation strategies a) Sequential b) Indexed c) Linked
5. Implement all File Organization Techniques a) Single level directory b) Two level c) Hierarchical d) DAG
6. Implement Bankers Algorithm for Dead Lock Avoidance
7. Implement all page replacement algorithms a) FIFO b) LRU c) LFU

**Cycle - II**

### Inter Process Communication and Process Synchronization

1. File system related system calls. (Learn to create, open, read, write, seek into, close files; open, read, write, search, close directories)
2. Process management – Fork, Exec (Learn to create a new process and to overlay an executable binary image on an existing process)
3. Inter-process communication between related processes using pipes.
4. Inter-process communication unrelated processes using Shared memory.

**Total contact Hours:** -  
**Total Tutorials:** -  
**Total Practical Classes:** 45  
**Total Hours:** 45
<table>
<thead>
<tr>
<th><strong>Subject Code</strong></th>
<th><strong>Subject</strong></th>
<th><strong>Hours / Week</strong></th>
<th><strong>Credit</strong></th>
<th><strong>Maximum Marks</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>CA157</td>
<td>Design and Analysis of Algorithms</td>
<td>3 L 1 T -</td>
<td>4 C</td>
<td>40 CA 60 SE 100 TM</td>
</tr>
</tbody>
</table>

**Prerequisite**
- To Introduce Problem Solving and design techniques
- To analyze the asymptotic performance of algorithms
- To apply important algorithmic design paradigms

**Objectives**
- On successful completion of the course, students will be able to:
  - Select appropriate algorithm techniques for real time problems
  - Analyze algorithm efficiency and complexity

**UNIT – I  Introduction to Algorithmic Analysis**
Definitions and notations: Standard notations – asymptotic notations – worst case, best case and average case analysis; big oh, small oh, omega and theta notations; Recursive algorithms, analysis of non-recursive and recursive algorithms, solving recurrence equations, analyzing control structures. Heap, shell, radix, insertion, selection and bubble sort; sequential, binary and Fibonacci search.

**UNIT – II  Divide and Conquer and Greedy method**

**UNIT – III  Dynamic Programming and Basic Traversal Techniques**

**UNIT – IV  Backtracking**

**UNIT – V  Branch and Bound Method**

**Total contact Hours: 45**  **Total Tutorials: 15**  **Total Practical Classes: -**  **Total Hours: 60**

**Text Books:**

**Reference Books:**

**Websites:**
1. http://www.cse.unt.edu
2. http://nptel.iitm.ac.in
<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA158</td>
<td>Object Oriented Programming</td>
<td>3</td>
<td>1</td>
<td>-</td>
</tr>
</tbody>
</table>

**Prerequisite**
- 

**Objectives**
- To introduce the basics of Object Oriented Programming.
- To educate the programming skill in C++ and Java.
- To practice the object oriented concepts using C++ and Java.

**Outcomes**
On successful completion of the course, students will be able to:
- Understand the Object Oriented Programming Concepts.
- Design and develop real world problem using C++ and Java.

**UNIT – I**
Limitations in structured programming - Characteristics of Object Oriented Language - Data types - Loops - Functions - Classes Objects - Constructors and Destructors - Operator Overloading and Type Conversion.

**UNIT – II**

**UNIT – III**
Java vs. C++ - Introduction to Java - Java on the Internet - Java Data types - Java keywords and Flow Control - Methods - Polymorphism - Exception handling - Multithreading - Persistence - Garbage collection.

**UNIT – IV**
Final declaration - Packages - Interfaces and Inner Class - Java I/O System - Run time type identification - User Interface design basics with swing.

**UNIT – V**
Network programming - Applets class - Architecture - Applet Programs - Abstract window tool kit.

**Total contact Hours:** 45  **Total Tutorials:** 15  **Total Practical Classes:** -  **Total Hours:** 60

**Text Books:**

**Reference Books:**

**Websites:**
1. http://nptel.ac.in
<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA159</td>
<td>Computer Organization and Architecture</td>
<td>3 1 -</td>
<td>4 40</td>
<td>60 100</td>
</tr>
</tbody>
</table>

**Prerequisite**

- To provide an overview of computer hardware and software.
- To give a methodical treatment of machine instructions, addressing techniques, and instruction sequencing.
- To explain the basics of I/O data transfer synchronization and a series of complex data structures.

**Objectives**

- Gain knowledge regarding the ways for increasing main memory bandwidth.
- Understand Processor implementation by both hardwired and Micro programmed control.

**Outcomes**

On successful completion of the course, students will be able to:
- Gain knowledge regarding the ways for increasing main memory bandwidth.
- Understand Processor implementation by both hardwired and Micro programmed control.

**UNIT – I Basic Computer Organization and Design**

<table>
<thead>
<tr>
<th>Instruction Codes</th>
<th>Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>09</td>
</tr>
</tbody>
</table>

**UNIT – II Micro-programmed Control and Processor Organization**

<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>09</td>
</tr>
</tbody>
</table>

**UNIT – III Memory Organization**

<table>
<thead>
<tr>
<th>Memory hierarchy – main memory – auxiliary memory – Associate memory – Cache memory – Virtual memory.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hours: 09</td>
</tr>
</tbody>
</table>

**UNIT – IV Input-Output Organization**

<table>
<thead>
<tr>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Hours: 09</td>
</tr>
</tbody>
</table>

**UNIT – V Parallel Processing**

<table>
<thead>
<tr>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Hours: 09</td>
</tr>
</tbody>
</table>

**Text Books:**


**Reference Books:**


**Websites:**

1. http://www.nptel.iitm.ac.in
2. http://dspace.utamu.ac.ug
**Department**: Computer Science and Engineering  
**Programme**: Master of Computer Application  
**Semester**: Two  
**Category**: TY  

<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>L</th>
<th>T</th>
<th>P</th>
<th>C</th>
<th>CA</th>
<th>SE</th>
<th>TM</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA160</td>
<td>Business Process</td>
<td>3</td>
<td>1</td>
<td>-</td>
<td>-</td>
<td>4</td>
<td>60</td>
<td></td>
<td>40</td>
<td>100</td>
</tr>
</tbody>
</table>

**Prerequisite**  
- 

**Objectives**  
- To introduce the core business process management concepts as applied to a variety of organizational processes and workflows.  
- To understand the role of business process management in operations improvement strategies and to understand the role of organizational culture and change management during business process improvement.

**Outcomes**  
On successful completion of the course, students will be able to:  
- Understand the basics concepts of business processes using their key operations characteristics  
- Have the ability to understand the fundamental of business process management (BPM) and its relationships with Total Quality Management (TQM), Business Process Reengineering (BPR) and Enterprise Resource Planning (ERP).

**UNIT – I**  
**Business Organizations**  
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Hours: 09</td>
<td></td>
</tr>
</tbody>
</table>

**UNIT – II**  
**Business Organization and Process Model**  
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Hours: 09</td>
<td></td>
</tr>
</tbody>
</table>

**UNIT – III**  
**Business Process and Total Quality**  
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Hours: 09</td>
<td></td>
</tr>
</tbody>
</table>

**UNIT – VI**  
**Business Process Reengineering**  
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Hours: 09</td>
<td></td>
</tr>
</tbody>
</table>

**UNIT – V**  
**Introduction to e-Business**  
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Hours: 09</td>
<td></td>
</tr>
</tbody>
</table>

**Total contact Hours**: 45  
**Total Tutorials**: 15  
**Total Practical Classes**: -  
**Total Hours**: 60

**Text Books:**  

**Reference Books:**  

**Websites:** -
<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA161</td>
<td>Computer Graphics and Animation</td>
<td>3 1 - 4</td>
<td>40</td>
<td>60 100</td>
</tr>
</tbody>
</table>

**Prerequisite**
- 

**Objectives**
- To learn Graphical devices and use them for developing software applications
- To learn, develop, design and implement two dimensional graphical structures.
- To understand the components of Graphics and Animation applications.
- To design innovative applications such as animation.

**Outcomes**
- On successful completion of this course
  - The students will get acquainted Graphics domains.
  - They will understand major intricacies of Graphics systems and animation applications.
  - They will be able to convert verbal descriptions to animations and vice versa.

**UNIT – I**

**Graphics Systems and Graphical User Interface**


**UNIT – II**

**Display Primitives and Transformations**


**UNIT – III**

**2D Transformations and Viewing**


**UNIT – IV**

**3D Concepts Representations and Transformations**


**UNIT – V**

**Computer Animation**


**Text Books:**

**Reference Books:**

**Websites:**
1. http://nptel.ac.in/courses/106106090/
2. http://nptel.ac.in/courses/106105032/
**Department:** Computer Science and Engineering  
**Programme:** Master of Computer Application  
**Semester:** Two  
**Category:** LB

<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA162</td>
<td>Algorithms Laboratory</td>
<td>-</td>
<td>3</td>
<td>60</td>
</tr>
</tbody>
</table>

**Prerequisite:** Programming with C

**Objectives:**
- To design and implement algorithms of different problems with different algorithm techniques
- To analyze various algorithmic techniques

**Outcomes:**
- Selection of relevant algorithm technique and data structures for the given problems in terms of memory and run time efficiency.
- Have the analysis of different techniques.

**Cycle -I**  
**Hours:** 21

1. Implementation of sorting algorithms with analysis of time and space complexity.
2. Implementation of searching algorithms with analysis of time and space complexity.

**Cycle -II**  
**Hours:** 24

2. Implementation of Traversal techniques.
3. Implementation of Backtracking.
5. Solving NP-Complete problems (using heuristics).

**Total contact Hours:** -  
**Total Tutorials:** -  
**Total Practical Classes:** 45  
**Total Hours:** 45
<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA163</td>
<td>Object Oriented Programming</td>
<td>-</td>
<td>3</td>
<td>60</td>
</tr>
<tr>
<td></td>
<td>Laboratory</td>
<td>2</td>
<td></td>
<td>40</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>100</td>
</tr>
</tbody>
</table>

**Prerequisite**: C Programming

**Objectives**
- To introduce the basics of Object Oriented Programming.
- To educate the programming skill in C++ and Java.
- To practice the object oriented concepts using C++ and Java.

**Outcomes**
On successful completion of the course, students will be able to:
- Understand the Object Oriented Programming Concepts.
- Have the ability to design and develop real world problem using C++ and Java.

**Cycle – I**

<table>
<thead>
<tr>
<th>C++ Programming</th>
<th>Hours: 21</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Program to implement Class and Object</td>
</tr>
<tr>
<td>2.</td>
<td>Program to implement Constructor Overloading</td>
</tr>
<tr>
<td>3.</td>
<td>Program to implement Inheritance</td>
</tr>
<tr>
<td>a.</td>
<td>Single Inheritance</td>
</tr>
<tr>
<td>b.</td>
<td>Multiple Inheritance</td>
</tr>
<tr>
<td>c.</td>
<td>Multi-level Inheritance</td>
</tr>
<tr>
<td>d.</td>
<td>Hybrid Inheritance</td>
</tr>
<tr>
<td>4.</td>
<td>Program to implement Polymorphism</td>
</tr>
<tr>
<td>5.</td>
<td>Program to implement Virtual Function</td>
</tr>
<tr>
<td>6.</td>
<td>Program to implement Operator Overloading Function</td>
</tr>
<tr>
<td>7.</td>
<td>Program to implement Templates</td>
</tr>
<tr>
<td>8.</td>
<td>Program to implement Exception Handling</td>
</tr>
</tbody>
</table>

**Cycle –II**

<table>
<thead>
<tr>
<th>JAVA Programming</th>
<th>Hours: 24</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Program to implement Class and Object</td>
</tr>
<tr>
<td>2.</td>
<td>Program to implement Constructor Overloading</td>
</tr>
<tr>
<td>3.</td>
<td>Program to implement Inheritance</td>
</tr>
<tr>
<td>a.</td>
<td>Single Inheritance</td>
</tr>
<tr>
<td>b.</td>
<td>Multi-level Inheritance</td>
</tr>
<tr>
<td>c.</td>
<td>Hybrid Inheritance</td>
</tr>
<tr>
<td>4.</td>
<td>Program to implement Multithreading</td>
</tr>
<tr>
<td>5.</td>
<td>Program to implement Exception Handling</td>
</tr>
<tr>
<td>6.</td>
<td>Program to implement Java Swing</td>
</tr>
<tr>
<td>7.</td>
<td>Program to implement Applets</td>
</tr>
<tr>
<td>8.</td>
<td>Network Programming</td>
</tr>
</tbody>
</table>

**Total contact Hours:** -
**Total Tutorials:** -
**Total Practical Classes:** 45
**Total Hours:** 45
## Objectives
- Understand the role of a database management system in an organization.
- Understand basic database concepts, including the structure and operation of the relational data model.
- Understand and successfully apply logical database design principles, including E-R diagrams and database normalization.
- Construct simple and moderately advanced database queries using Structured Query Language (SQL).

## Outcome
On successful completion of the course, students will be able to:
- Gain knowledge regarding the design, management and manipulate the databases
- Able to design the various applications that includes Databases and able to manage the data effectively.

### UNIT – I
Introduction of Database Systems and E-R Model

<table>
<thead>
<tr>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>L T P C CA SE TM</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3 1 - 4 40 60 100</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>


### UNIT – II
Relational Model and SQL

<table>
<thead>
<tr>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>L T P C CA SE TM</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3 1 - 4 40 60 100</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>


### UNIT – III
Integrity, Security and Relational Database Design

<table>
<thead>
<tr>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>L T P C CA SE TM</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3 1 - 4 40 60 100</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>


### UNIT – IV
Storage, File Structures, Indexing and Hashing

<table>
<thead>
<tr>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>L T P C CA SE TM</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3 1 - 4 40 60 100</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>


### UNIT – V
Transactions and Concurrency Control

<table>
<thead>
<tr>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>L T P C CA SE TM</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3 1 - 4 40 60 100</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>


<table>
<thead>
<tr>
<th>Total contact Hours: 45</th>
<th>Total Tutorials: 15</th>
<th>Total Practical Classes: -</th>
<th>Total Hours: 60</th>
</tr>
</thead>
</table>

### Text Books:

### Reference Books:

### Websites:
<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA165</td>
<td>Computer Networks</td>
<td>L 3 T 1 P -</td>
<td>4</td>
<td>CA 40 SE 60 TM 100</td>
</tr>
</tbody>
</table>

**Prerequisite**

- To familiarize the students with the basic taxonomy and terminology of networks
- To introduce networks technologies, application protocols, e-mail and communication protocols.

**Objectives**

- To understand the details and functionality of layered network architecture.
- To identify the different types of network topologies and protocols.

**Outcomes**

On successful completion of the course, students will be able to:

- Understand the details and functionality of layered network architecture.
- Identify the different types of network topologies and protocols.

**UNIT – I**  
Introduction to Computer Networks  
Hours: 09


**UNIT – II**  
Data Link Layer  
Hours: 09


**UNIT – III**  
Network Layer  
Hours: 09

Network Layer Design Issues - Routing Algorithms - Quality of Service - Internetworking.

**UNIT – IV**  
Transport Layer and Presentation Layer  
Hours: 09


**UNIT – V**  
Application Layer  
Hours: 09


**Text Books:**


**Reference Books:**


**Websites:**

1. [http://www.tutorialspoint.com/computer_fundamentals/computer_networking.htm](http://www.tutorialspoint.com/computer_fundamentals/computer_networking.htm)
2. [http://nptel.ac.in/video.php?subjectId=106105081](http://nptel.ac.in/video.php?subjectId=106105081)
Department : Computer Science and Engineering
Programme : Master of Computer Application

Semester : Three
Category : TCM

<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Course Name</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA166</td>
<td>Platform Technologies</td>
<td>3 - 2 - 4</td>
<td>50</td>
<td>50</td>
</tr>
</tbody>
</table>

Prerequisite -

Objectives
- To understand the foundations of CLR execution.
- To know the object oriented aspects of C#.
- To learn web based applications on .NET (ASP.NET).

Outcomes
On successful completion of the course, students will be able to:
- Gain knowledge of application development using .NET.
- Design and develop Web based applications on .NET.

UNIT – I
Introduction to C#
Hours: 09
Introducing C#, Understanding .NET, overview of C#, Literals, Variables, Data Types, Operators, checked and unchecked operators, Expressions, Branching, Looping, Methods, implicit and explicit casting, Constant, Arrays, Array Class, Array List, String, String Builder, Structure, Enumerations, boxing and unboxing.

UNIT – II
Object Oriented Aspects of C#
Hours: 09
Class, Objects, Constructors and its types, inheritance, properties, indexers, index overloading, polymorphism, sealed class and methods, interface, abstract class, abstract and interface, operator overloading, delegates, events, errors and exception, Threading.

UNIT – III
Application Development on .NET
Hours: 09
Building windows application, Creating our own window forms with events and controls, menu creation, inheriting window forms, SDI and MDI application, Dialog Box(Modal and Modeless), accessing data with ADO.NET, DataSet, typed dataset, Data Adapter, updating database using stored procedures, SQL Server with ADO.NET, handling exceptions, validating controls, windows application configuration.

UNIT – IV
Web Based Application Development on .NET
Hours: 09
Programming web application with web forms, ASP.NET introduction, working with XML and .NET, Creating Virtual Directory and Web Application, session management techniques, web.config, web services, passing datasets, returning datasets from web services, handling transaction, handling exceptions, returning exceptions from SQL Server.

UNIT – V
CLR and .NET Framework
Hours: 09
Assemblies, Versioning, Attributes, reflection, viewing meta data, type discovery, reflection on type, marshalling, remoting, security in .NET

Mini Project : Hours:30
The students are to be made of batches of two or three members in a team. Each team should select an appropriate project, which should be implemented in .NET framework. At the end of the project each team should prepare a report which consists of at least the following. Abstract, Introduction, Problem Statement, Design Document, Results, Interpretation of the Results

Total contact Hours: 45
Total Tutorials: -
Total Practical Classes: 30
Total Hours: 75

Text Books:

Reference Books:

Websites:
1. www.w3.org/standards
<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA167</td>
<td>Database Management Systems Laboratory</td>
<td>-</td>
<td>3</td>
<td>60</td>
</tr>
</tbody>
</table>

Prerequisite: -

Objectives:
- To understand basic concepts in all advanced aspects as well as to get some practical hand-on experience with commercial database management systems
- To design and implement database application system

Outcomes:
On successful completion of the course, students will be able to:
- Design databases using data modeling and data normalization techniques
- Create databases using popular database management system products
- Solve problems by constructing database queries using the Structured Query Language
- Develop insights into future data management tool and technique trends

**Cycle - I**  
**Hours: 30**

2. Study of SQL: Primitive Data Types – User Defined data Types – Built-in Functions – Parts of Speech of create, alter, drop, select, insert, delete, update, commit, rollback, save point, grant, revoke.
4. Study of Procedural Query Language: Blocks, Exception Handling, Functions, Procedures, Cursors, Triggers, Packages

**Cycle - II**  
**Hours: 15**

1. Application: Design and develop any two of the following:
   a. Library Information System
   b. Logistics Management System
   c. Students’ Information System
   d. Ticket Reservation System
   e. Hotel Management System
   f. Hospital Management System
   g. Inventory Control
   h. Retail Shop Management
   i. Employee Information System
   j. Payroll System
   k. Any other Similar System

Total contact Hours: -  
Total Tutorials: -  
Total Practical Classes: 45  
Total Hours: 45
Department: Computer Science and Engineering  
Programme: Master of Computer Application

<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA168</td>
<td>Computer Networks Laboratory</td>
<td>L T P C C A  SE TM</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Prerequisite:
- To give them hands-on experience on networking
- To gain knowledge about routing protocols and encryption algorithms.
- To practice performance evaluation of network protocols

Objectives:
- On successful completion of the course, students will be able to:
  - Understand Network protocols
  - Encryption Algorithms will create security awareness among students
  - Develop services that are required for network management.

Outcomes:
- Implementation of peer to peer communication using UDP
- Implementation of socket program for UDP Echo Client and Echo Server
- Implementation of Client Server Communication Using TCP
- Implementation of CRC
- Message passing using Message Window
- Implementation of port scanner
- Implementation of Sliding Window Protocol
- Create a socket using HTTP for web page upload and download.

Cycle - I  
Hours: 24

1. Implementation of peer to peer communication using UDP
2. Implementation of socket program for UDP Echo Client and Echo Server
3. Implementation of Client Server Communication Using TCP
4. Implementation of CRC
5. Message passing using Message Window
6. Implementation of port scanner
7. Implementation of Sliding Window Protocol
8. Create a socket using HTTP for web page upload and download.

Cycle - I  
Hours: 21

1. Create a socket using HTTP for web page upload and download.
2. Implementation of Subnetting
3. Implement a Routing Protocol like DSR / AODV for transmitting up of data between sender and receiver (Using Network Simulators like NS2 or Glomosim)
4. Implementation of RPC
5. Implementation of DES
6. Implementation of RSA
7. Implementation of image Steganography
8. Implementation of Email

Total contact Hours: -  
Total Tutorials: -  
Total Practical Classes: 45  
Total Hours: 45
<table>
<thead>
<tr>
<th>Semester: Three</th>
<th>Programme: Master of Computer Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>Subject Code: HS151</td>
<td>Subject: Communication Skills</td>
</tr>
<tr>
<td>Prerequisite: -</td>
<td></td>
</tr>
<tr>
<td>Hours / Week</td>
<td>Credit</td>
</tr>
<tr>
<td>L</td>
<td>T</td>
</tr>
<tr>
<td>2</td>
<td>-</td>
</tr>
</tbody>
</table>

### Objectives
- To assist the students to develop their communication skills and communicate with confidence.
- To enhance the student’s employability prospects by sharpening their skills.

### Outcomes
On successful completion of the course, students will be able to:
- Interact with ease without any inhibitions.
- Acquire the much needed confidence to converse in English fluently.
- Imbibe the requisite skills for facing the interview and group discussion.

### Text Books

### Reference Books

### Websites:

---

**Department**: Humanities and Social Sciences

**Programme**: Master of Computer Application

<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>HS151</td>
<td>Communication Skills</td>
<td>2</td>
<td>-</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td></td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3</td>
<td>1</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

---

**Total contact Hours**: 45

**Total Tutorials**: -

**Total Practical Classes**: 3

**Total Hours**: 45
Department: Computer Science and Engineering  
Programme: Master of Computer Application  
Semester: Four  
Category: TY

<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA169</td>
<td>Web Technologies</td>
<td>3 L 1 P -</td>
<td>4 C</td>
<td>40 CA 60 SE 100 TM</td>
</tr>
</tbody>
</table>

Prerequisite: -

Objectives:
- To learn web programming languages features.
- To understand the major components of Internet and associated protocols.
- To design innovative applications for web.
- To familiarize the latest web technologies.

Outcomes:
On successful completion of the course, students will be able to:
- Acquaint with client side and server side programming languages for web.
- Understand the major components and protocols of Internet application.
- Design web applications and services using latest technologies.

UNIT – I  
Internet Protocols, HTML 5.0 and JavaScript  
Hours: 09


UNIT – II  
Servlets and JSP  
Hours: 09


UNIT – III  
PHP and XML  
Hours: 09

An introduction to PHP: PHP- Using PHP- Variables- Program control- Built-in functions-Connecting to Database – JSON(basics) - XML: Basic XML- Document Type Definition- XML Schema, DOM.

UNIT – IV  
Multimedia and E-Business  
Hours: 09


UNIT – V  
Introduction To AJAX And Web Services  
Hours: 09

AJAX: Ajax Client Server Architecture; Web Services: Introduction to Web Services, UDDI, SOAP, WSDL, Web Service Architecture, Developing and deploying web services.

Total contact Hours: 45  
Total Tutorials: 15  
Total Practical Classes: --  
Total Hours: 60

Text Books:

Reference Books:
3. Ron Schmelzer, Travis Vandersypen, Jason Bloomberg, MadhuSiddalingaiah, Sam hunting, MichealD.Qualls, David Houlding, Chad Darby, Diane Kennedy, XML and Web Services, Sams, February, 2002.

Websites:
1. www.webtechnologies.co.tz
2. www.w3schools.com
<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA170</td>
<td>Theory of Computation</td>
<td>3</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>40</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>60</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>100</td>
</tr>
</tbody>
</table>

**Prerequisite**

- Present the theory of finite automata, as the first step towards learning advanced topics, such as compiler design.
- Discussing the applications of finite automata towards text processing.
- Develop an understanding of computation.

**Objectives**

On successful completion of the course, students will be able to:

- Develop a clear understanding of problem solvability and undecidability.
- Understand the equivalence between Non-deterministic Finite State Automata and Deterministic Finite State Automata.
- Understand the design and implementation of Lexical Analyzer and Parser.
- Appreciate the power of the Turing Machine, as an abstract automaton, that describes computation, effectively and efficiently.

**UNIT – I**

**Finite Automata and Regular Expressions**

Formal Languages and Regular expressions, Regular Expressions in programming languages, Deterministic and Non-Deterministic Finite Automata, Finite Automata with ε-moves, Conversion of NFA and DFA, Minimization of Finite Automata, Applications of Finite Automata, Implementation of Lexical Analyzer.

**UNIT – II**

**Regular Sets and Context Free Grammars**


**UNIT – III**

**Pushdown Automata**

Introduction on pushdown automata, Deterministic and Nondeterministic pushdown automata, Acceptance by PDA, Equivalence between pushdown automata and context-free grammars, Applications of pushdown automata.

**UNIT – IV**

**Parser**


**UNIT – V**

**Turing machines**


**Text Books:**


**Reference Books:**


**Websites:**

1. http://nptel.ac.in/courses/106106049/
2. http://nptel.ac.in/courses/106103070/
### Software Engineering

<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA171</td>
<td>Software Engineering</td>
<td>3</td>
<td>1</td>
<td>40</td>
</tr>
</tbody>
</table>

#### Prerequisite
- None

#### Objectives
- To provide information about wider engineering issues that form the background to develop complex, evolving (software-intensive) systems.
- To plan a software engineering process to account for functional and non-functional requirements.
- To impart knowledge to translate requirement specifications into a design, and then realize that design practically, all using an appropriate software engineering methodology.

#### Outcomes
On successful completion of the module students will be able to:
- Choose an appropriate process model for designing a project.
- Analyze the stakeholders’ requirements and develops suitable models.
- Demonstrate the knowledge of technologies and standards for designing a suitable project.
- Test the functional and nonfunctional aspects of the project.

### UNIT – I


**Hours: 09**

### UNIT – II


**Hours: 09**

### UNIT – III


**Hours: 09**

### UNIT – IV


**Hours: 09**

### UNIT – V


**Total contact Hours: 45**
**Total Tutorials: 15**
**Total Practical Classes: -**
**Total Hours: 60**

### Text Books:

### Reference Books:

### Websites:
1. [http://nptel.ac.in/courses/106101061/](http://nptel.ac.in/courses/106101061/)
### Subject: Web Technologies Laboratory

**Subject Code:** CA172

<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA172</td>
<td>Web Technologies Laboratory</td>
<td>-</td>
<td>3</td>
<td>60</td>
</tr>
</tbody>
</table>

**Prerequisite:**
- To learn the evolution of web and its associated technologies.
- To get familiarity on Client side and server side scripting.
- To provide hands on experience on building web application.

**Objectives:**
- On successful completion of the course, students will be able to:
  - Design and develop web applications.
  - Incorporate recent technologies such as Ajax and web services for developing e-commerce applications.

**Cycle - I**

**HTML, CSS Client Side Scripting**

<table>
<thead>
<tr>
<th>Hours: 21</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Creation of HTML Files</td>
</tr>
<tr>
<td>2. Working with Cascading Style Sheets and DOM</td>
</tr>
<tr>
<td>3. Working with Client Side Scripting</td>
</tr>
<tr>
<td>4. Experiments in Java Servlets</td>
</tr>
<tr>
<td>5. Experiments in Ajax Programming</td>
</tr>
</tbody>
</table>

**Cycle - II**

**Servlets and JSP**

<table>
<thead>
<tr>
<th>Hours: 24</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Working with other Server Side Scripting.</td>
</tr>
<tr>
<td>2. Developing Web Applications using XML.</td>
</tr>
<tr>
<td>5. Developing any E-commerce application using JSP (Mini Project)</td>
</tr>
</tbody>
</table>

**Total contact Hours:** -

**Total Tutorials:** -

**Total Practical Classes:** 45

**Total Hours:** 45
<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA173</td>
<td>CASE Tools Laboratory</td>
<td>-</td>
<td>3</td>
<td>60</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Prerequisite</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Objectives</th>
</tr>
</thead>
<tbody>
<tr>
<td>To understand the software engineering methodologies for project development.</td>
</tr>
<tr>
<td>To gain knowledge about open source tools for Computer Aided Software Engineering.</td>
</tr>
<tr>
<td>To develop an efficient software using case tools</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Outcomes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Upon Completion of the course, the students should be able to:</td>
</tr>
<tr>
<td>• Prepare an SRS for the Sample project.</td>
</tr>
<tr>
<td>• Draw following model diagrams: Use Case and Activity Diagrams Sequence,</td>
</tr>
<tr>
<td>Collaboration and Class Diagram using CASE Tools</td>
</tr>
<tr>
<td>• Perform Code coverage testing, memory leaks and other tests using CASEtools</td>
</tr>
</tbody>
</table>

SOFTWARE REQUIRED: Open source Tools: StarUML / UMLGraph / Topcased Prepare the following documents for each experiment and develop the software using software engineering methodology.

1. **Problem Analysis and Project Planning** - Thorough study of the problem – Identify Project scope - Objectives and Infrastructure
2. **Modeling** - Use work products – data dictionary - use case diagrams and activity diagrams - build and test class diagrams - sequence diagrams and add interface to class diagrams.
3. **Software Development and Debugging** – implement the design by coding
4. **Software Testing** - Prepare test plan - perform validation testing - coverage analysis - memory leaks - develop test case hierarchy - Site check and site monitor.

**Sample Experiments**

**Academic Domain**
1. Online objective type test examination system
2. Course Registration System under Autonomous System

**Transport domain**
3. Online ticket reservation system for PTDC Buses
4. GPS based bus monitoring system

**Finance domain**
5. ATM system
6. Share buying and Selling software

**Human Resource Management**
7. E-mail Client system
8. Staff scheduling System in a project

**Total contact Hours:** -  **Total Tutorials:** -  **Total Practical Classes:** 45  **Total Hours:** 45
## Mobile Computing

<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA174</td>
<td>Mobile Computing</td>
<td>3 1 -</td>
<td>4</td>
<td>40 60 100</td>
</tr>
</tbody>
</table>

### Prerequisite
- Computer Networks

### Objectives
- To introduce the basic concepts of mobile communications.
- To understand the wireless protocols, network layer and transport layer functionalities.
- Understand concepts of ad hoc and sensor networks.
- To impart the knowledge about mobile application development using android.

### Outcomes
- Develop a clear understanding of mobile network standards.
- Understand clearly challenges and solutions of wireless communications.
- Able to develop android based applications.

### UNIT – I Wireless Communications overview

### UNIT – II Mobile Short Range Networks

### UNIT – III Mobile Network and Transport Layer

### UNIT – IV Mobile Ad Hoc and Sensor Networks

### UNIT – V Mobile Application Development

### Text Books:

### Reference Books:

### Websites:
### Management Concepts and Strategies

<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA175</td>
<td>Management Concepts and Strategies</td>
<td>4 - - -</td>
<td>4</td>
<td>40</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>CA</td>
<td>60</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>SE</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>TM</td>
<td></td>
</tr>
</tbody>
</table>

**Prerequisite:**
- 

**Objectives:**
- To understand the different aspects of management
- To have the knowledge of general management principles
- To understand the challenges in managing organization and people
- To understand the need to control and monitor operations with suitable methods

**Outcomes:**
On successful completion of the course students will be able to:
- Realize the need for management
- Plan strategies and policies for organizations
- Perform the role human resource manager
- Monitor and Implement changes in operations for improving efficiency

**UNIT – I**
**Planning**


**UNIT – II**
**Organizing**

Decision making - The Nature and purpose of organizing - Basic departmentation - Line / staff Authority and decentralization - Effective Organizing and organizational culture.

**UNIT – III**
**Staffing**

Human Resource Management and selection - Performance appraisal and career strategy - Manager and organizational development.

**UNIT – IV**
**Leading**

Managing and the Human factor - Motivation - Leadership - communication.

**UNIT – V**
**Controlling**

The system and Process of controlling - control techniques and information Technology - Productivity and Operations Management - Overall and Preventive Control - Towards a unified, Global management theory.

**Total contact Hours:** 45  
**Total Tutorials:** 15  
**Total Practical Classes:**  
**Total Hours:** 60

### Text Books:

### Reference Books:

### Websites:
<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA176</td>
<td>Mobile Application Development Laboratory</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

**Subject Details**

**Semester:** Five  
**Category:** LB  
**Department:** Computer Science and Engineering  
**Programme:** Master of Computer Application

<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>L</th>
<th>T</th>
<th>P</th>
<th>C</th>
<th>CA</th>
<th>SE</th>
<th>TM</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA176</td>
<td>Mobile Application Development Laboratory</td>
<td></td>
<td></td>
<td>3</td>
<td>2</td>
<td>60</td>
<td>40</td>
<td>100</td>
</tr>
</tbody>
</table>

**Prerequisite:** -

**Objectives**

The student should be made to:
- Know the components and structure of mobile application development frameworks for Android and windows OS based mobiles.
- Understand how to work with various mobile application development frameworks.
- Learn the basic and important design concepts and issues of development of mobile applications.

**Outcome**

On successful completion of this course
- The student should be able to: Design and Implement various mobile applications using emulators.
- Deploy applications to hand-held devices

**Cycle - I**

1. Develop an application that uses GUI components, Font and Colours
2. Develop an application that uses Layout Managers and event listeners.
3. Develop a native calculator application.
4. Write an application that draws basic graphical primitives on the screen.
5. Develop an application that makes use of database.
6. Develop an application that makes use of RSS Feed.

**Hours:** 18

**Cycle - II**

1. Implement an application that implements Multi threading
2. Develop a native application that uses GPS location information.
3. Implement an application that writes data to the SD card.
4. Implement an application that creates an alert upon receiving a message.
5. Write a mobile application that creates alarm clock

**Hours:** 27

**Total contact Hours:** -  
**Total Tutorials:** -  
**Total Practical Classes:** 45  
**Total Hours:** 45
<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA177</td>
<td>Mini Project</td>
<td>-</td>
<td>3</td>
<td>60</td>
</tr>
</tbody>
</table>

**Objective**

- The objective of the mini project is to make the students to apply theoretical and practical tools/techniques to solve real life problems related to industry and academic institutions.

**Outcomes**

- After the completion of this project work the student should be able to:
  - Describe the Systems Development Life Cycle (SDLC).
  - Determine how to collect information to determine requirements and to design SRS
  - Generate reports
  - To decide the future scope and further enhancement of the system

The students are expected to develop application oriented small scale projects, through which students explore their technical skills. The students shall carry out the project assigned to them and submit a report at the end of the semester for continuous assessment.

<p>| Total Contact Hours: - | Total Tutorials: - | Total Practical Classes: 45 | Total Hours: 45 |</p>
<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA178</td>
<td>Comprehensive Test and Viva-Voce</td>
<td>-</td>
<td>3</td>
<td>60</td>
</tr>
</tbody>
</table>

**Prerequisite**: -

**Objectives**
- To refresh the subjects studied.
- To face the placement tests conducted for the campus recruitment with a sound fundamental knowledge.

**Outcomes**
- Take up competitive exams for higher studies.
- Able to confidently appear placement interviews.

The students are provided with tutorial sessions to update and refresh their knowledge in all courses they have studied from the 1st to 5th semester. Two comprehensive tests, preferably with objective type questions from above courses will be conducted. A comprehensive viva voce examination (external) will be conducted.

<table>
<thead>
<tr>
<th>Total Contact Hours: -</th>
<th>Total Tutorials: -</th>
<th>Total Practical Classes: 45</th>
<th>Total Hours: 45</th>
</tr>
</thead>
</table>

**Text Books**: 1. Test Books prescribed for the courses in the respective syllabus from I to V semester.
<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA179</td>
<td>Project Work</td>
<td>L T P C</td>
<td>CA SE TM</td>
<td>10 150 150 100</td>
</tr>
</tbody>
</table>

**Prerequisite**
- 

**Objectives**
- To expose the students with project-product development cycle using state-of-art technologies.
- To understand the Product Development Cycle
- To plan for various activities of the project.
- To acquire in depth working knowledge in the chosen area of problem

**Outcomes**
- Acquire knowledge and skills needed for the construction of software project namely, design, development, testing and documentation phases
- Enhance the technical presentation skills.

**Project Work**
1. Select a Real time Application/ Industrial problem.
2. Perform a feasibility study.
3. Study the limitations of the Existing System.
4. Define the Problem Statement and Objectives.
5. Detailed Design of the Proposed Solution.
6. Finalize the Experimental Environment.
7. Implementation of the Proposed Solution.

| Total contact Hours: - | Total Tutorials: - | Total Practical Classes: - | Total Hours: - |
SYLLABUS (Elective Subjects)
Department : Computer Science and Engineering  
Programme : Master of Computer Application

<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAE51</td>
<td>Object Oriented Analysis and Design</td>
<td>3</td>
<td>-</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>40</td>
<td>60</td>
<td>100</td>
</tr>
</tbody>
</table>

**Prerequisite**
- To understand the object oriented life cycle.
- To know how to identify objects, relationships, services and attributes through UML.
- To understand the use-case diagrams.
- To know the Object Oriented Design process.
- To know about software quality and usability.

**Objectives**
On successful completion of the course, students will be able to:
- Differentiate object oriented analysis design from structured design.
- Design a software development process up to industry standard.

**Outcomes**
- Social and Ethical issues and challenges in software development.
- To design an object-oriented system using UML.
- To develop object-oriented software using existing CASE tools.

**UNIT – I**
Introduction


**UNIT – II**
Objects and Classes

*Basics Object classes Relationships between classes - The class diagram Stages in building a class diagram - Packages -Using the class diagram in system development.*

**UNIT – III**
Identifying functionality

*Introduction CRC cards and interaction diagrams - Identifying operations using the CRC card technique - Interaction diagrams Specifying operations - Using the CRC cards and interaction diagrams in system development - State Diagrams - States and events Constructing a state diagram - Using state diagrams in system development.*

**UNIT – IV**
Activity Diagrams

*Introduction - Modeling a sequence of activities - Modeling alternative courses of action - modeling iteration of activities- Modeling activities that are carried out in parallel -Swim lanes Design Architecture Implementation diagrams -The user interface Dealing with persistent data.*

**UNIT – V**
Designing Objects and Classes


| Total contact Hours: 45 | Total Tutorials: - | Total Practical Classes: - | Total Hours: 45 |

**Text Books:**

**Reference Books:**

**Websites:**
<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAES2</td>
<td>System Software</td>
<td>3</td>
<td>-</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>40</td>
<td>60</td>
</tr>
</tbody>
</table>

**Prerequisite**

**Objectives**

- To design assemblers, loaders, linkers
- To understand system software support program execution
- To understand the data structures required to implement assemblers, loaders and linkers

**Outcomes**

- To realize how assembling and linking of programs are carried out
- To compare and appreciate how translation could be improved by efficiently through good programming style

**UNIT – I**

**Machine Architecture and Introduction to Assembler**


**UNIT – II**

**Assembler and Macroprocessor**


**UNIT – III**

**Loaders and Linkers**


**UNIT – IV**

**Compilers**


**UNIT – V**

**Editors and Debugging Systems**

Text Editors - Overview of Editing Process, User Interface, Editor Structure, Interactive Debugging Systems - Debugging Functions and Capabilities, Relationship With Other Parts Of The System, User-Interface Criteria

**Text Books:**


**Reference Books:**


**Websites:**

<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAE53</td>
<td>Information Security</td>
<td>3</td>
<td>-</td>
<td>40</td>
</tr>
</tbody>
</table>

| Prerequisite | -                      |

### Objectives
- To introduce the need for information security in the computer-driven world.
- To introduce the security techniques for the protection of information system
- To familiarize the concepts of security planning, Laws and Risk management strategies.

### Outcomes
On successful completion of the course, the students will be able to:
- Identify the need for information security and the various threats and attacks against the CIA triad of information security.
- Identify the various types of security controls available to protect the information infrastructure.
- Perform effective information security risk management.

#### UNIT – I
**Introduction**


#### UNIT – II
**Planning for Security**


#### UNIT – III
**Security Technology**


#### UNIT – IV
**Risk Management**

Overview, Risk identification, Risk Assessment, Risk Control Strategies, Quantitative vs. Qualitative Risk Control Practices.

#### UNIT – V
**Implementing Information Security**


Total contact Hours: 45

Text Books:

Reference Books:

Websites:
2. http://liscs.wssu.edu/drupal/node/2991
<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAES4</td>
<td>Cloud Computing</td>
<td>3</td>
<td>40</td>
<td>60</td>
</tr>
</tbody>
</table>

**Prerequisite**

- To understand the concept of virtualization, set up a private cloud
- To learn and apply different cloud programming models as per need
- To learn the design and implementation of cloud services.

**Objectives**

- Understand the cloud computing architectures and models
- Apply suitable virtualization concept, choose the appropriate cloud service provider, programming models and setup a private cloud

**Outcomes**

- Understand the cloud computing architectures and models
- Apply suitable virtualization concept, choose the appropriate cloud service provider, programming models and setup a private cloud

**UNIT – I**

**Cloud Computing Architecture and Model**

- Cloud ecosystem – Service management – Computing on demand

**UNIT – II**

**Virtual Machine**


**UNIT – III**

**Cloud Infrastructure**


**UNIT – IV**

**Programming Model and SLA Management**

- Hadoop and MapRuduce – Cloud Application Design – SLA Management in Cloud Computing : A Service provider’s Perspective - Big data Analytics

**UNIT – V**

**Cloud Security**


**Text Books:**


**Reference Books:**


**Websites :**

1. http://nptel.ac.in
2. http://india.emc.com
Department: Computer Science and Engineering
Programme: Master of Computer Application

<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>L</th>
<th>T</th>
<th>P</th>
<th>C</th>
<th>CA</th>
<th>SE</th>
<th>TM</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAE55</td>
<td>Hardware and Troubleshooting</td>
<td>3</td>
<td>-</td>
<td>-</td>
<td>3</td>
<td>40</td>
<td>60</td>
<td>100</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Prerequisite

- To study the fundamentals of PC hardware and Peripherals.
- To understand the working principles of hardware devices and components.
- To understand the system resources and their uses.
- To bridge the gap between the theoretical study of Computer Organization and the practical study of the hardware components in use.
- To practice the troubleshooting of hardware and network bugs in real life.

Objectives

On successful completion of the course, students will be able to:

- Map the theoretical concepts of Computer Organization and Microprocessors to the real-life Personal Computer organization.
- Interface the PC to the real-time application entities of their own engineering discipline.
- Troubleshoot any kind of systems and fix networking bugs in practice.

Outcomes

UNIT – I

<table>
<thead>
<tr>
<th>Hours: 09</th>
</tr>
</thead>
</table>


UNIT – II

<table>
<thead>
<tr>
<th>Hours: 09</th>
</tr>
</thead>
</table>


UNIT – III

<table>
<thead>
<tr>
<th>Hours: 09</th>
</tr>
</thead>
</table>


UNIT – IV

<table>
<thead>
<tr>
<th>Hours: 09</th>
</tr>
</thead>
</table>


UNIT – V

<table>
<thead>
<tr>
<th>Hours: 09</th>
</tr>
</thead>
</table>


Total contact Hours: 45
Total Tutorials: -
Total Practical Classes: -
Total Hours: 45

Text Books:

Reference Books:
Websites:
2. http://pcsupport.about.com
Department : Computer Science and Engineering
Programme : Master of Computer Application

Semester : -
Category : TY

<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAE56</td>
<td>Accounting and Financial Management</td>
<td>3 - - - 3</td>
<td>40 60 100</td>
<td></td>
</tr>
</tbody>
</table>

Prerequisite

Objectives
- To introduce the basics terminologies and concepts of economics.
- To educate more analysis methods related economics.
- To impart the role of managers and various management concepts in organizations.

Outcomes
On successful completion of the course, students will be able to:
- Understand the basics of concepts, principles and practical applications of Economics, which govern the functioning of a firm/organization under different market conditions
- Have the ability to understand the fundamental concepts and principles of management; the basic roles, skills, functions of management and various organizational structures

UNIT – I
Introduction to Management: Definition, Principles, Nature and Scope – Role and function of Managers – Levels of Management – Manager vs Leader – Functions of Management (Planning, Organizing, Staffing, Directing and Controlling) – Types of Management (Finance, Marketing, Human Resource, Production)

UNIT – II

UNIT – III

UNIT – VI

UNIT – V

Total contact Hours: 45  Total Tutorials: -  Total Practical Classes: -  Total Hours: 45

Text Books:

Reference Books:

Websites: -
Department: Computer Science and Engineering  
Programme: Master of Computer Application

Semester: -  
Category: TY

<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAE57</td>
<td>Resource Management Techniques</td>
<td>3</td>
<td>-</td>
<td>40</td>
</tr>
</tbody>
</table>

Prerequisite:

Objectives:
- To understand the basics of operations research
- To have the knowledge of mathematical aspects of resource management
- To understand the challenges in decision making
- To understand the issues in scheduling

Outcomes:
On successful completion of the course students will be able to:
- Implement strategies for resource management and scheduling
- Understand risks and uncertainty in decision making
- Adopt job scheduling principles in production
- Manage Inventory

UNIT – I  
Transportation Problem  
Hours: 09

UNIT – II  
Inventory Control  
Hours: 09

UNIT – III  
Production Scheduling  
Hours: 09
Introduction - single machine scheduling - flow shop scheduling - job shop scheduling.

UNIT – IV  
Decision Theory  
Hours: 09

UNIT – V  
Queuing Theory  
Hours: 09

Total contact Hours: 45  
Total Tutorials: -  
Total Practical Classes: -  
Total Hours: 45

Text Books:

Reference Books:

Websites:
1. http://people.brunel.ac.uk/~mastjjb/jeb/or/contents.html
2. http://www.slideshare.net/hemanthcrpatna/operations-research-6215649
<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAE58</td>
<td>Agent Technologies</td>
<td>3</td>
<td>-</td>
<td>40</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>3</td>
<td>60</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>100</td>
</tr>
</tbody>
</table>

**Prerequisite**: 
- To understand the concept of agents and multi–agent systems and the main applications for which they are appropriate
- To know the typical applications for agent technology
- To understand deductive reasoning agents, practical reasoning agents, reactive and hybrid agents

**On successful completion of the course students will be able to:**
- Have knowledge on an intelligent software agent and its characteristics
- Describe multi-agent learning, and mobile agent characteristics and applications
- Design multi-agent system by using knowledge representation technique, and agent communication and coordination mechanisms

**UNIT – I**
Hours: 09


**UNIT – II**
Hours: 09


**UNIT – III**
Hours: 09


**UNIT – IV**
Hours: 09


**UNIT – V**
Hours: 09


Total contact Hours: 45

Text Books:

Reference Books:

Websites:
### Subject: UNIX Internals

<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAE59</td>
<td>UNIX Internals</td>
<td>3 - - -</td>
<td>3</td>
<td>40 60 100</td>
</tr>
</tbody>
</table>

#### Prerequisite
- To learn UNIX environment, UNIX shells and Important utilities in UNIX
- To gain knowledge about Networking Utilities, System programming system calls, files, processes, sockets and pipes and memory management.

#### Objectives
- On successful completion of the module students will be able to:
  - Design the necessary file system based on the requirement
  - Write system level programs such as file recover, process interaction etc.
  - Design a proper communication among the various processes in a multi process system.

#### Outcomes

#### UNIT – I
**Introduction to the Kernel and File sub-system**
- Architecture of the UNIX operating system – Introduction to the system concepts – Kernel Data Structures; The Buffer Cache: Buffer Headers – Structure – Retrieval of a buffer – Reading and writing disk blocks – Advantages and Disadvantages; Internal Representation of Files: Inode – Structure of a regular file – Directories – Conversion of a pathname to an Inode – Super Block – Inode Assignment

#### UNIT – II
**System Calls for the file system**

#### UNIT – III
**Process Control**

#### UNIT – IV
**Memory Management Policies**
- Swapping – Demand Paging – A Hybrid System with swapping and demand paging; The I/O Subsystem: Drier Interfaces, Disk Drivers, Terminal Drivers, Streams.

#### UNIT – V
**Inter Process Communication**

#### Total contact Hours: 45  Total Tutorials: -  Total Practical Classes: -  Total Hours: 45

#### Text Books:

#### Reference Books:

#### Websites:
1. http://www.ee.surrey.ac.uk/Teaching/Unix/
<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Course Name</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAE60</td>
<td>Software Architecture</td>
<td>3</td>
<td>40</td>
<td>60</td>
</tr>
</tbody>
</table>

**Prerequisite**

- To learn how to develop quality software products
- To introduce fundamentals of software design
- To understand the software architecture evaluation process

**Objectives**

- Have acquired traits of a good software architect
- Able to design software products that ensures quality
- Able to choose from a set of alternative designs

**UNIT – I**

Hours: 09

Characteristics of design activities – Elements of design – Software quality models and their effects – Quality Attributes – Basic rules of software design – Design process.

**UNIT – II**

Hours: 09


**UNIT – III**

Hours: 09

Choices of styles and their combination – Hierarchical styles – Simultaneously heterogeneous style – Locationally heterogeneous style

**UNIT – IV**

Hours: 09

Case Study - Key word frequency vector - Theory of design spaces – Design Space of architectural elements – Design Space of architectural styles.

**UNIT – V**

Hours: 09


**Total contact Hours: 45**

**Total Tutorials:** -

**Total Practical Classes:** -

**Total Hours:** 45

**Text Books:**


**Reference Books:**


**Websites:**

## Subject Code: CAE61

### Subject: Social Network Analysis

<table>
<thead>
<tr>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>L</td>
<td>T</td>
<td>P</td>
</tr>
<tr>
<td>3</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

### Prerequisite

- The students are to
  - Be exposed to the overview of social networks and its importance.
  - Understand the social network concepts and various methods of analysis.
  - Understand the underlying mathematical structures of Social Networks.
  - Get an overview of the structural properties of Social Networks.

### Objectives

On successful completion of the course, students will be able to:
- Understand the theories and concepts of social networks.
- Understand the data in the Social Network way.
- Identify, formulate and analyze Social network problems.
- Interpret the results obtained.

### Outcomes

<table>
<thead>
<tr>
<th>UNIT – I</th>
<th>Network, Relations and Structure</th>
<th>Hours: 09</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Social Networks Perspective- Network Data- Boundary Specification and Sampling- Types of Networks-Network Data, Measurement and Collection</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>UNIT – II</th>
<th>Mathematical Representations of Social Networks</th>
<th>Hours: 09</th>
</tr>
</thead>
<tbody>
<tr>
<td>Graph Theoretic Notation- Sociometric Notation- Algebraic Notation- Graphs- Directed Graphs- Signed Graphs-Signed Directed Graphs- Valued Graphs- Valued Directed Graphs- Multi Graphs- Hyper Graphs- Relations-Matrices- Properties</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>UNIT – III</th>
<th>Structural and Locational Properties</th>
<th>Hours: 09</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actor Centrality- Degree Centrality- Closeness Centrality- Betweenness Centrality- Information Centrality- Structural Balance- Clusterability- Generalizations of Clusterability- Transitivity</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>UNIT – IV</th>
<th>Roles and Positions</th>
<th>Hours: 09</th>
</tr>
</thead>
<tbody>
<tr>
<td>Background- Structural Equivalence- Automorphic and Isomorphic Equivalence- Regular Equivalence- Types of Ties- Local Role Equivalence- Ego Algebras</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>UNIT – V</th>
<th>DYADIC and TRIADIC Methods</th>
<th>Hours: 09</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Dyad Census- Examples - An Index for Mutuality- Simple Distributions on Digraphs- Conditional Uniform Distributions- The Triad Census- Examples - Mean and Variance of a Triad Census</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Total contact Hours: 45</th>
<th>Total Tutorials: -</th>
<th>Total Practical Classes: -</th>
<th>Total Hours: 45</th>
</tr>
</thead>
</table>

### Text Books:

### Reference Books:

### Websites:
<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAE62</td>
<td>Advanced Java Programming</td>
<td>3 - - 3</td>
<td>40</td>
<td>60 100</td>
</tr>
</tbody>
</table>

**Prerequisite**

- To learn advanced Java programming concepts like Swings, Networking, Beans, etc.
- To develop network and distributed applications in Java
- To understand issues in enterprise applications development
- To learn Java messaging and J2ME

**Objectives**

- Develop Swing-based GUI
- Develop client/server applications and socket programming concepts
- Design, Update and retrieve the data from the databases using SQL
- Develop distributed applications using RMI and component-based Java software using JavaBeans
- Develop server side programs in the form of Servlets and enterprise applications.

**UNIT – I**

Advanced Swing graphical interface components – Model view controller - Java I/O streaming – filter and pipe streams – Threading.  
**Hours: 09**

**UNIT – II**

**Hours: 09**

**UNIT – III**

Server side programming – servlets – Java Server Pages – JSTL - Applet to Applet communication – applet to Servlet communication - JDBC – Applications on databases  
**Hours: 09**

**UNIT – IV**

Multimedia streaming applications – Java Media Framework –Enterprise Java beans component model –Session EJB & distributed transactions – Entity EJB  
**Hours: 09**

**UNIT – V**

Messaging with JMS – Wireless application development and J2ME – Enterprise Java Case Study – Architecture – presentation and Controller logic – Business logic  
**Hours: 09**

**Total contact Hours: 45**  
**Total Tutorials:**  
**Total Practical Classes:**  
**Total Hours: 45**

**Text Books:**


**Reference Books:**


**Websites:**

1. http://sites.google.com/site/advancedjavabooksandppts
2. http://docs.oracle.com/javase/7/docs/api/
<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAE63</td>
<td>Data Mining and Warehousing</td>
<td>3</td>
<td>100</td>
</tr>
</tbody>
</table>

**Prerequisite**

- To introduce Data Preprocessing, Architecture and basic concepts of data mining
- To understand the Association rule mining techniques
- To describe and demonstrating basic data mining algorithms, methods, and tools
- To familiarize with clustering and classification algorithms
- To understand the overall architecture of a data warehouse techniques and Hardware and Operational design methods

**Objectives**

On successful completion of the course, students will be able to:

- Design applications related to Association rule mining algorithm
- Analyze uses of different classification algorithms for better classification
- Design various clustering techniques and solve outlier problems
- Design the architecture of Data warehousing methods
- Implement the data mining, data warehousing techniques and methods in integrating and interpreting datasets

**UNIT – I**

Introduction

Preprocessing, Cleaning, Integration, Transformation, Reduction, Discretization, Concept Hierarchy Generation, Data Mining Primitives, Query Language, Graphical User Interfaces, Architectures, Concept Description, Data Generalization, Characterizations, Class Comparisons, Descriptive Statistical Measures.

**UNIT – II**

Association Rule Mining

Association Rule Mining, Single-Dimensional Boolean Association Rules from Transactional Databases, Multi-Level Association Rules from Transaction Databases-mining multidimensional Association rules – association mining to correlation analysis-constraint based association mining.

**UNIT – III**

Classification and Prediction

Classification and Prediction, Issues, Decision Tree Induction, Bayesian Classification, Association Rule Based, Other Classification Methods, Prediction, Classifier Accuracy.

**UNIT – IV**

Cluster Analysis


**UNIT – V**

Data Warehousing

Data warehouse architecture and design, Hardware and Operational design, Tuning and testing.

**Text Books:**


**Reference Books:**


**Websites:**

1. http://web.engr.illinois.edu/~hanj/bk2/datamining concepts
3. http://datawarehouse4u.info/
**Department**: Computer Science and Engineering  
**Programme**: Master of Computer Application  
**Semester**: -  
**Category**: TY

<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAE64</td>
<td>Artificial Intelligence</td>
<td>3</td>
<td>3</td>
<td>100</td>
</tr>
</tbody>
</table>

**Prerequisite**
- To search and discover intelligent characteristics of existing AI projects, intelligent agents map a new problem – as search.
- To understand different search strategies for a problem.
- To understand different Knowledge Representation schemes for typical AI problems.
- To design and implement a typical AI problem to be solved Using Machine Learning Techniques.

**Objectives**

- To search and discover intelligent characteristics of existing AI projects, intelligent agents map a new problem – as search.
- To understand different search strategies for a problem.
- To understand different Knowledge Representation schemes for typical AI problems.
- To design and implement a typical AI problem to be solved using Machine Learning Techniques.

**Outcome**

On successful completion of this course students will be able to:
- Develop intelligent systems
- Design applications related to Natural Language Processing and Web applications.

**UNIT – I**  
**Introduction**  
Hours: 09


**UNIT – II**  
**Knowledge Representation**  
Hours: 09

Approaches and issues in knowledge representation- Propositional Logic –Predicate logic-Forward and backward reasoning - Unification- Resolution- Weak slot-filler structure – Strong slot-filler structure- Knowledge-Based Agent

**UNIT – III**  
**Reasoning under uncertainty**  
Hours: 09


**UNIT – IV**  
**Planning and Learning**  
Hours: 09

Planning with state space search-partial order planning-planning graphs-conditional planning-continuous planning- Multi-Agent planning. Forms of learning- Learning from observation - Inductive learning – Decision trees – Explanation based learning – Statistical Learning methods - Reinforcement Learning - Neural Net learning and Genetic learning

**UNIT – V**  
**Advanced Topics**  
Hours: 09


**Total Contact Hours**: 45  
**Total Tutorials**: -  
**Total Practical Classes**: -  
**Total Hours**: 45

**Text Books**:

**Reference Books**:

**Websites**:
2. www.stanford.edu/class/cs221/
<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAE65</td>
<td>Principles of Distributed Systems</td>
<td>L</td>
<td>T</td>
<td>P</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Prerequisite:

- To introduce the Architecture, and types of Distributed Systems
- To Learn the Fundamental Principles Distributed Systems
- To Learn the Design Issues of the distributed Systems
- To Learn the Design of Fault Tolerant Systems

Objectives:

On successful completion of the course, students will be able to:

- Design Middleware Components for Distributed Systems
- Create a Distributed System through the integration of Heterogeneous Applications and Web services
- Demonstrate the understanding of need for distributed systems and their applications
- Design of Fault Tolerant Systems

Outcomes:

UNIT – I

Distributed Systems Architecture and Operating Systems


UNIT – II

Remote Procedure Call

Introduction – RPC model – Transparency of RPC- Implementing RPC mechanism- Stub generation- Marshaling arguments and results- Server management- Parameter passing semantics – Call semantics- Communication protocols for RPCs – Complicated RPC client server binding- Exception handling- Security- Special types of RPCs-RPCs in heterogeneous environments- Lightweight RPC

UNIT – III

Distributed Shared Memory and Synchronization

Introduction – General architecture of DSM systems- Design and implementation of DSM- Granularity- Structure of shared memory space- Consistency models- Replacement strategy- Thrashing- Advantages of DSM – Synchronization- Clock synchronization- Event ordering- Mutual exclusion- Deadlock- Election Algorithm

UNIT – IV

Resource and Process Management


UNIT – V

Distributed File Systems


Total contact Hours: 45
Total Tutorials: -
Total Practical Classes:-
Total Hours: 45

Text Books:


Reference Books:


Websites:

1. http://nptel.ac.in
<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAE66</td>
<td>Distributed Database System</td>
<td>L 3 T - P -</td>
<td>C 3</td>
<td>CA 40 CA 60</td>
</tr>
</tbody>
</table>

Prerequisite

- To introduce the basics of distributed database.
- To educate various database management techniques.
- To allow computer system resources to be used in an efficient manner by using database model.
- Makes the computer more convenient to access the various data

Objectives

- On successful completion of the course, students will be able to:
  - Understand the basics concepts of distributed database and its related techniques.
  - Provide a pleasant and effective user interface

Outcomes

UNIT – I Distributed Data Processing Hours: 09
Introduction-Distributed data processing, distributed database design, distributed query processing, distributed directory management, distributed concurrency control, distributed deadlock management, reliability of distributed DBMS, operating system support, heterogeneous databases. Overview of Relational DBMS.

UNIT – II Distributed DBMS Architecture Hours: 09

UNIT – III Query Processing in Distributed Databases Hours: 09
Overview of Query Processing – Query Decomposition and Data Localization – Optimization of Distributed Queries, Multi-database Query Processing.

UNIT – IV Deadlock management Hours: 09
Deadlock prevention, deadlock avoidance, deadlock detection and resolution. Distributed DBMS Reliability: System, state and failure, reliability and availability, mean time between failures/mean time to report, failure and fault tolerance in distributed systems: reasons for failures, basic fault tolerance approaches and techniques.

UNIT – V Transaction Management And other Advanced Systems Hours: 09

Total contact Hours: 45 Total Tutorials:-- Total Practical Classes: - Total Hours:45

Text Books:

Reference Books:

Websites:
1. http://docs.oracle.com/cd/B10501_01/server.920/a96521/ds_concepts.htm
**Department**: Computer Science and Engineering  
**Programme**: Master of Computer Application

<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAE67</td>
<td>Software Testing and Quality Assurance</td>
<td>3</td>
<td>-</td>
<td>100</td>
</tr>
</tbody>
</table>

**Prerequisite**

- To understand the theoretical aspects of software testing
- To have the knowledge of the existing testing methods
- To understand the challenges in applying testing and analysis methods in software development and maintenance
- To emphasis on software quality measurement, and quality standards

**Objectives**

On successful completion of the course students will be able to:

- Detect, classify, prevent and remove defects
- Conduct formal inspections, record and evaluate results of inspections
- Choose and design appropriate testing strategies and develop test cases
- Analyze the software quality with software quality standards

**Outcomes**

**UNIT – I**  
**Software Testing Fundamentals**  
Hours: 09


**UNIT – II**  
**Testing Types**  
Hours: 09

White Box, Black Box and Grey Box – White box testing techniques - Statement coverage, Branch Coverage, Condition coverage, Decision/Condition coverage, Multiple condition coverage, Dataflow coverage, Mutation testing – Black box testing techniques – Boundary value analysis, Equivalence partitioning, Syntax testing, Finite state testing.

**UNIT – III**  
**Testing Object Oriented Software**  
Hours: 09


**UNIT – IV**  
**Software Quality**  
Hours: 09


**UNIT – V**  
**Software Quality Standards**  
Hours: 09


**Text Books:**


**Reference Books:**


**Websites:**

1. www.softwaretestinghelp.com/
2. www.softwaretestingfundamentals.com/
3. www.aptest.com/resources.html
Department : Computer Science and Engineering
Programme : Master of Computer Application

<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAE68</td>
<td>Software Project Management</td>
<td>3</td>
<td>40</td>
<td>100</td>
</tr>
</tbody>
</table>

Prerequisite

- To develop an awareness of the need for project planning and management
- To explain the procedures needed to monitor, control and report upon an IT development project
- To explain the ways in which appropriate quality attributes of the products of an IT development project can be assessed and assured

Objectives

- On completion of this course the student will able to
  - Apply the fundamentals of software project process framework
  - Do project planning, scheduling, tracking, organizing and controlling
  - Utilize the Quality management and tools used
  - Estimate cost and evaluate the project

Outcomes

- Introduction To Software Project Management
- Introduction - Software Project vs other types of Project
- Contract and Technical Project Management
- Activities - Plans, Methods and Methodologies
- Categorizing - Stake Holders
- Setting Objective
- Business Cases - Management Control
- Project Evaluation
- And Programme Management
- Introduction
- Business Case - Portfolio - Evaluation - Cost - Risk - Program Management
- Resource Allocation - Strategic - Program Creation - Aids - Reservation
- - Benefits - An Overview Of Project Planning - Introduction
- step wise project planning
- 10 steps.

UNIT – I

Selection Of An Appropriate Project Approach
- Introduction - Build Or Buy
- Choosing Methodologies and Technologies
- Process Model - Structure vs Speed
- Of Delivery - Waterfall and Spiral Model
- Prototyping - Categories
- Incremental Delivery - Agile - Extreme Programming
- Managing Iterative Process
- Process Model Selection - Software Effort Estimation
- Estimates - Over And Under Estimates - Software Estimating Basis
- Effort Estimation Technique
- Bottom Up - Top Down Approach
- Expert Judgment - Analogy
- Albrecht Function Point
- Mark II - COSMIC-Cocomo
- Activity Planning - Objective
- Plan - Project Schedule - Activities - Sequencing
- And Scheduling - Network Planning
- Models - Formulating - Time Dimension
- Forward And Backward Pass
- Activity Float
- Identifying Critical Path
- And Activities
- Shortening Project Duration
- Activity On Arrow Network

UNIT – II

Risk Management
- Introduction - Risk Categories - Framework - Identification
- Assessment - Planning - Management
- Evaluation - Pert Technique
- Monte Carlo
- Critical Chain
- Resource Allocation
- Nature - Identifying Requirements
- Scheduling
- Creating Critical Path
- Counting the Cost
- Resource Schedule
- Cost Schedule - Scheduling Sequence
- Monitoring and Control
- Framework - Data Collection
- Visualizing - Cost Monitoring - Earned Value Analysis
- Prioritizing Monitoring
- Change Control

UNIT – III

Managing contracts
- Types - stages - Contract management
- Acceptance
- Managing people in software environments
- Understanding Behaviour
- organisation behavior
- selection - instruction - Motivation
- Oldham
- hackman Ethical and professional concern
- Working in teams
- Becoming team - decision making
- organizational structure
- coordination dependencies
- Dispersed and virtual teams
- Communication genres and plans - Leadership

UNIT – V

Software reliability
- statistical testing
- software quality
- software quality management system
- ISO 9000 - SEI Capability Maturity Model
- Personal Software Process - Six Sigma

Total contact Hours: 45
Total Tutorials: -
Total Practical Classes: -
Total Hours: 45

Text Books:


Reference Books:


Websites:
1. http://nptel.ac.in/courses/106101061/
Department : Computer Science and Engineering
Programme : Master of Computer Application

<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAE69</td>
<td>Parallel Computing</td>
<td>3</td>
<td>-</td>
<td>40</td>
</tr>
</tbody>
</table>

Prerequisite

Objectives

- To introduce parallel computing models, communication models, Parallel programming platforms and algorithms

Outcomes

- Students acquire the skills to implement software effectively and efficiently on parallel hardware platforms such as multi-core processors and processors that use multithreading techniques.

UNIT – I Introduction to Parallel Computing
Motivating Parallelism-Scope of parallel computing-Parallel programming platforms-Implicit Parallelism Limitations of Memory System Performance-Dichotomy of Parallel computing platforms-Physical organization of parallel platforms-Communication costs in parallel machines-Routing mechanisms for inter connection networks.

UNIT – II Principles of Parallel Algorithm Design
Preliminaries-Decomposition techniques-characteristics of tasks and interactions-mapping techniques for load balancing-methods for containing interaction overheads-parallel algorithm models.

UNIT – III Basic Communication Operations
One to all broadcast and all to one reduction-all to all broadcast and reduction-scatter and gather-sources of overhead in parallel programs-performance metrics for parallel systems-the effect of granularity on performance.

UNIT – IV Message Passing Paradigm
Principles of message passing programming-Building blocks-Message passing interface-Topologies and embedding-Overlapping computation with communication-Collective communication and computation operation.

UNIT – V Programming Shared Address Space Platforms

Total contact Hours: 45
Total Tutorials: -
Total Practical Classes: -
Total Hours: 45

Text Books:

Reference Books:

Websites:
1. http:// www.nptel.ac.in
# Big Data Analytics

**Department**: Compute Science and Engineering  
**Programme**: M.C.A  
**Semester**: -  
**Category**: TY  

<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Hours / Week</th>
<th>Credit</th>
<th>Maximum Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAE70</td>
<td>Big Data Analytics</td>
<td>3 - -</td>
<td>3</td>
<td>40 60 100</td>
</tr>
</tbody>
</table>

**Prerequisite**: -

**Objectives**:  
- To learn to analyze the big data using intelligent techniques.  
- To understand the various search methods and visualization techniques.  
- To learn to use various techniques for mining data stream.  
- To understand the applications using Map Reduce Concepts  

**Outcomes**: On completion of this course the student will able to  
- Analyze the big data analytic techniques for useful business applications.  
- Design efficient algorithms for mining the data from large volumes.  
- Analyze the HADOOP and Map Reduce technologies associated with big data analytics  
- Explore on Big Data applications Using Pig and Hive  

---

## UNIT – I  
### Introduction to Big Data  

**Hours**: 09

---

## UNIT – II  
### Mining Data Streams  

**Hours**: 09

---

## UNIT – III  
### Hadoop  
**History of Hadoop- The Hadoop Distributed File System – Components of Hadoop- Analyzing the Data with Hadoop- Scaling Out- Hadoop Streaming- Design of HDFS-Java interfaces to HDFS Basics- Developing a Map Reduce Application-How Map Reduce Works-Anatomy of a Map Reduce Job run-Failures-Job Scheduling-Shuffle and Sort – Task execution - Map Reduce Types and Formats- Map Reduce Features**

**Hours**: 09

---

## UNIT – IV  
### Hadoop Environment  

**Hours**: 09

---

## UNIT – V  
### Frameworks  
**Applications on Big Data Using Pig and Hive – Data processing operators in Pig – Hive services – HiveQL – Querying Data in Hive - fundamentals of HBase and ZooKeeper - IBM InfoSphereBigInsights and Streams. Visualizations - Visual data analysis techniques, interaction techniques; Systems and applications**

**Total contact Hours**: 45  
**Total Tutorials**: -  
**Total Practical Classes**: -  
**Total Hours**: 45

**Text Books**:


**Reference Books**:

3. Jiawei Han, MichelineKamber, Data Mining Concepts and Techniques, Second Edition,  
6. Michael Minelli (Author), Michele Chambers (Author), AmbigaDhiraj (Author) , Big Data, BigAnalytics:
Emerging Business Intelligence and Analytic Trends for Today's Businesses, Wiley Publications, 2013


**Websites:**

1. nptel.ac.in/syllabus/110105060
### Department: Computer Science and Engineering
#### Programme: Master of Computer Application

**Semester:**

**Subject Code** | **Subject** | **Hours / Week** | **Credit** | **Maximum Marks** | **L** | **T** | **P** | **C** | **CA** | **SE** | **TM**
---|---|---|---|---|---|---|---|---|---|---|---|---
CAE71 | User Interface Design | 3 | - | - | 3 | 40 | 60 | 100

**Prerequisite:**

- 

**Objective:**

- To develop specific skills, competencies, and points of view needed by professionals in user interface design and implementation

**Outcomes:**

- On successful completion of the course, students will be able to:
  - Identify and define key terms related to user interfaces and user interface design and implementation
  - Identify and describe various types of computer users and computer use contexts
  - Describe and explain the user interface design process

**UNIT – I**


**UNIT – II**


**UNIT – III**


**UNIT – VI**


**UNIT – V**


**Total contact Hours:** 45  
**Total Tutorials:** -  
**Total Practical Classes:** -  
**Total Hours:** 45

**Text Books:**


**Reference Books:**


**Websites:**

1. nptel.ac.in/courses/106105087/pdf/m09L20.pdf
2. https://www.iam.unibe.ch/scg/svn_repos/Lectures/ESE/08UIDesign.ppt
<table>
<thead>
<tr>
<th>Department: Computer Science and Engineering</th>
<th>Programme: Master of Computer Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>Semester:</td>
<td>Category: TY</td>
</tr>
<tr>
<td>Subject Code: CAE72</td>
<td>Hours / Week:</td>
</tr>
<tr>
<td>Subject: Multimedia Systems and Applications</td>
<td>Credit: C</td>
</tr>
<tr>
<td>Prerequisite:</td>
<td>Maximum Marks:</td>
</tr>
<tr>
<td></td>
<td>L</td>
</tr>
<tr>
<td></td>
<td>3</td>
</tr>
</tbody>
</table>

**Objectives**

- To introduce the principles and current technologies of multimedia systems, issues in effectively representing, processing, and retrieving multimedia data such as sound and music, graphics, image and video.
- To discuss latest Web technologies and some advanced topics in current multimedia research.

**Outcomes**

On successful completion of the course, the students will be able to:

- Understand the characteristics of different media.
- Gain knowledge about different compression principles.
- Program multimedia data and design and implement media applications.

**UNIT – I**

**Introduction to Multimedia Systems**


**UNIT – II**

**Data Compression**

Storage Space - Coding Requirements Source, Entropy and Hybrid Coding, Some Basic Compression Techniques - JPEG - H.261 (px64) - MPEG - DVI. Optical Storage Media: Basic Technology - Video Disks and Other Worms - Compact Disk Digital Audio - Compact Disk Read only Memory - CD-ROM Extended Architecture - Further CD-ROM-based Developments - Compact Disk Write Once - Compact Disk Magneto Optical - Computer Technology

**UNIT – III**

**Multimedia Operating Systems**


**UNIT – IV**

**Multimedia Database Management Systems**

Multimedia Database Management Systems - Characteristic of an MDBMS - Data Analysis - Data Structure - Operations on Data - Integration in a Database Model - Documents, Hypertext and MHEG - Documents and Documents Architecture - Manipulation of Multimedia Data - Hypertext and Hypermedia - Documents Architecture - SGML - Document Architecture - ODA - MHEG-User Interfaces - General Design Issues - Video at the User Interface - Audio at the User Interface - User-friendliness as the Primary Goal

**UNIT – V**

**Synchronization and Multimedia Applications**


**Text Books:**


**Reference Books:**

3. Yao Wang, Joern Ostermann, and Ya-Qin Zhang, Video Processing and Communications, Prentice Hall,

<table>
<thead>
<tr>
<th>Websites:</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. <a href="http://hsc.csu.edu.au/ipt/mm_systems/7-1/what_is_multimedia.htm">http://hsc.csu.edu.au/ipt/mm_systems/7-1/what_is_multimedia.htm</a></td>
</tr>
<tr>
<td>2. <a href="http://www.tutorialspoint.com/listtutorials/multimedia/1">http://www.tutorialspoint.com/listtutorials/multimedia/1</a></td>
</tr>
<tr>
<td>Subject Code</td>
</tr>
<tr>
<td>--------------</td>
</tr>
<tr>
<td>CAE73</td>
</tr>
</tbody>
</table>

**Prerequisite**

- To introduce the Architecture, Instruction Set and Programming of 8086 microprocessor
- To Learn the Peripheral Interfacing of Microprocessors
- To Learn the Architecture, Instruction Set, Programming and Interfacing of 8051 Microcontroller
- To impart through case studies, the system design principles using 8086 and 8051

**Objectives**

On successful completion of the course, students will be able to:

- Design and Implement Assembly Language Programs Using 8086 Microprocessor
- Interface Microprocessor and Microcontroller with different kinds of Peripherals
- Design and Implement Assembly Language Programs Using 8051 Microprocessor
- Design and Implement the Microprocessor and Microcontroller based Systems

**Outcomes**

UNIT – I  
**16-bit Microprocessor Architecture and Programming**  
**Hours: 09**


UNIT – II  
**Memory and I/O Interfacing**  
**Hours: 09**

Memory and I/O interfacing – Interrupts – Assembly Language Programming - string manipulation, array operations.

UNIT – III  
**Peripheral Devices and Interfacing**  
**Hours: 09**

Peripheral Devices and Interfacing – Parallel Data Communication Interface – Serial Data Communication Interface – Keyboard Display Interface – ADC and DAC Interface – Programmable Timer Interface

UNIT – IV  
**8086 Microprocessor-based System Design**  
**Hours: 09**


UNIT – V  
**8-bit Microcontroller**  
**Hours: 09**


Total contact Hours: 45  
Total Tutorials: -  
Total Practical Classes: -  
Total Hours: 45

**Text Books:**


**Reference Books:**


**Websites:**

1. http://nptel.ac.in
2. http://www.intel.in