Rashtreeya Sikshana Samithi Trust

R.V. College of Engineering

(Autonomous Institution affiliated to VTU, Belagavi)

Master of Computer Applications
(M.C.A)

Scheme and Syllabus

2016
Vision:

Pioneering in ICT Enabled Quality Education and Research with a focus on Sustainable and Inclusive Applications

Mission

- To adapt novel methodologies for quality education through experiential learning
- To empower students with continuous, holistic education, emphasizing on discipline, ethics and social commitment
- To become a vibrant knowledge center for research and software development
- To continuously build capacity steering towards industry-institute collaborative research and entrepreneurial competencies
- To utilize and develop free and open source software tools for sustainable and inclusive growth

Program Educational Objectives (PEO)

MCA graduates will be able to:

PEO1: Practice software engineering principles and standards to develop software to meet customer requirements across verticals

PEO2: Contribute to build sustainable and inclusive applications using mathematical, simulation and meta-heuristic models

PEO3: Demonstrate entrepreneurial qualities through individual competence and teamwork

PEO4: Achieve successful professional career with integrity and societal commitments leading to lifelong learning

Program Outcomes (PO)

MCA graduates will be able to:

PO1: **Computational Knowledge:** Acquire in-depth computational knowledge and mathematics with an ability to abstract and conceptualise models from defined problems and requirements

PO2: **Problem Analysis:** Identify, formulate, conduct literature survey and solve complex computing problems through analysis as well as provide optimal solutions

PO3: **Design / Development of Solutions:** Design and evaluate solutions for complex problems, components or processes that meet specified needs after considering public health and safety, cultural, societal, and environmental factors
PO4: **Conduct investigations of complex Computing problems:** Conduct literature survey to analyse and extract information relevant to unfamiliar problems and synthesise information to provide valid conclusions and interpret data by applying appropriate research methods, tools and design experiments

PO5: **Modern Tool Usage:** Create, select, adapt and apply appropriate techniques, resources and modern IT tools to complex computing system activities, with an understanding of the limitations

PO6: **Professional Ethics:** Understand and commit to professional ethics and cyber regulations, responsibilities, and norms of professional computing practices

PO7: **Life-long Learning:** Engage in lifelong learning independently for continual development to improve knowledge and competence as a computing professional

PO8: **Project management and finance:** Demonstrate knowledge and understanding of management principles and apply these to multidisciplinary software development as a team member and manage projects efficiently as a leader considering economical and financial factors

PO9: **Communication Efficacy:** Understand and communicate effectively with the computing community and with society at large, regarding complex computing systems activities confidently and effectively by writing effective reports and design documentations by adhering to appropriate standards, make effective presentations and give / receive clear instructions

PO10: **Societal and Environmental Concern:** Understand responsibilities and consequences based on societal, environmental, health, safety, legal and cultural issues within local and global contexts relevant to professional computing practices

PO11: **Individual and Team Work:** Function effectively as an individual, as a member or leader in diverse teams in multidisciplinary environments

PO12: **Innovation and Entrepreneurship:** Identify a timely opportunity for entrepreneurship and use innovation to pursue and create value addition for the betterment of the individual and society at large

**Program Specific Outcomes (PSO)**

MCA graduates will be able to:

PSO1: Solve real world computing system problems of various industries by understanding and applying the principles of mathematics, computing techniques and business concepts

PSO2: Design, test, develop and maintain desktop, web, mobile and cross platform software applications using modern tools and technologies
## FIRST SEMESTER

<table>
<thead>
<tr>
<th>Sl. No</th>
<th>Course Code</th>
<th>Course Title</th>
<th>BoS</th>
<th>Credit Allocation</th>
<th>Total Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Lecture</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Tutorial</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Practical</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Experiential</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Learning</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>16MCA11</td>
<td>Data Structures (Theory and Practice)</td>
<td>MCA</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>16MCA12</td>
<td>Operating Systems (Theory and Practice)</td>
<td>MCA</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>16MCA13</td>
<td>Web Technologies (Theory and Practice)</td>
<td>MCA</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>16MCA14</td>
<td>Computer Organization and Architecture</td>
<td>MCA</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>16MAT15</td>
<td>Discrete Mathematics</td>
<td>MAT</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Contact (Hrs/ week)</td>
<td>25</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Non Contact (Hrs/ week)*</td>
<td>09</td>
</tr>
</tbody>
</table>

### SECOND SEMESTER

<table>
<thead>
<tr>
<th>Sl. No</th>
<th>Course Code</th>
<th>Course Title</th>
<th>BoS</th>
<th>Credit Allocation</th>
<th>Total Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Lecture</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Tutorial</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Practical</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Experiential</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Learning</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>16MCA21</td>
<td>Database Systems (Theory and Practice)</td>
<td>MCA</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>16MCA22</td>
<td>Object Oriented Programming (Theory and Practice)</td>
<td>MCA</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>16MCA23</td>
<td>Analysis and Design of Algorithms (Theory and Practice)</td>
<td>MCA</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>16MCA24</td>
<td>Software Engineering</td>
<td>MCA</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>16MCA25</td>
<td>Management Information System and e-Commerce</td>
<td>MCA</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Contact (Hrs/ week)</td>
<td>25</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Non Contact (Hrs/ week)*</td>
<td>26</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Total: 19 1 3 2 25

Total: 19 1 6 0 26

Total: 0 1 0 8 --
### THIRD SEMESTER

<table>
<thead>
<tr>
<th>Sl. No</th>
<th>Course Code</th>
<th>Course Title</th>
<th>BoS</th>
<th>Credit Allocation</th>
<th>Total Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Lecture</td>
<td>Tutorial</td>
</tr>
<tr>
<td>1</td>
<td>16MCA31</td>
<td>Computer Networks (Theory and Practice)</td>
<td>MCA</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>16MCA32</td>
<td>Software Testing and Practices (Theory and Practice)</td>
<td>MCA</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>16MCA33X</td>
<td>Elective-1 (Theory and Practice)</td>
<td>MCA</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>16MCA34X</td>
<td>Elective-2</td>
<td>MCA</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>16MCA35</td>
<td>Research Methodology</td>
<td>MCA</td>
<td>3</td>
<td>0</td>
</tr>
</tbody>
</table>

| Contact (Hrs/ week) | 19  | 0  | 3 | 2 | 24 |
| NonContact (Hrs/week)* | 0  | 0  | 6 | 0 | 25 |

**Elective 1**

(5 Credits) Theory + Practice

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Title</th>
<th>Course Code</th>
<th>Course Title</th>
<th>Course Code</th>
<th>Course Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>16MCA331</td>
<td>Content Management System</td>
<td>16MCA332</td>
<td>Advanced Object Oriented Programming</td>
<td>16MCA333</td>
<td>Model view controller Programming</td>
</tr>
</tbody>
</table>

**Elective 2**

(5 Credits)

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Title</th>
<th>Course Code</th>
<th>Course Title</th>
<th>Course Code</th>
<th>Course Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>16MCA341</td>
<td>System Programming</td>
<td>16MCA342</td>
<td>Advanced Database Systems</td>
<td>16MCA343</td>
<td>Operations Research</td>
</tr>
</tbody>
</table>
## FOURTH SEMESTER

<table>
<thead>
<tr>
<th>Sl. No</th>
<th>Course Code</th>
<th>Course Title</th>
<th>BoS</th>
<th>Credit Allocation</th>
<th>Total Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Lecture</td>
<td>Tutorial</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Pratical</td>
<td>Experiential Learning</td>
</tr>
<tr>
<td>1</td>
<td>16MCA41</td>
<td>Enterprise Application Programming (Theory and Practice)</td>
<td>MCA</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>16MCA42X</td>
<td>Elective-3 (Theory and Practice)</td>
<td>MCA</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>16MCA43X</td>
<td>Elective-4</td>
<td>MCA</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>16MCA44</td>
<td>Software Project Management</td>
<td>MCA</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>16MCA45</td>
<td>Business Communication</td>
<td>MCA</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>16MCA46</td>
<td>Minor project-1</td>
<td>MCA</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Total</td>
<td></td>
<td>18</td>
<td>1</td>
</tr>
</tbody>
</table>

Contact (Hrs/week) 18 1 9 0 28
Noncontact (Hrs/week) 0 1 5 8 13

**Elective 3**

(5 Credits) Theory + Practice

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Title</th>
<th>Course Code</th>
<th>Course Title</th>
<th>Course Code</th>
<th>Course Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>16MCA42</td>
<td>Advanced Computer Networks</td>
<td>16MCA422</td>
<td>Mobile Application Development</td>
<td>16MCA423</td>
<td>Computer Graphics</td>
</tr>
</tbody>
</table>

**Elective 4**

(5 Credits)

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Title</th>
<th>Course Code</th>
<th>Course Title</th>
<th>Course Code</th>
<th>Course Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>16MCA43</td>
<td>Network Security</td>
<td>16MCA432</td>
<td>Information Retrieval</td>
<td>16MCA433</td>
<td>Modeling and Simulation</td>
</tr>
</tbody>
</table>
# R. V. College of Engineering, Bengaluru – 59.
(An Autonomous Institution affiliated to VTU, Belagavi)

## Department of Master of Computer Applications

### FIFTH SEMESTER

<table>
<thead>
<tr>
<th>Sl. No</th>
<th>Course Code</th>
<th>Course Title</th>
<th>BoS</th>
<th>CREDIT ALLOCATION</th>
<th>Total Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Lecture L</td>
<td>Tutorial T</td>
</tr>
<tr>
<td>1</td>
<td>16MCA51</td>
<td>Data Analytics (Theory and Practice)</td>
<td>MCA</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>16MCA52X</td>
<td>Elective - 5 (Theory and Practice)</td>
<td>MCA</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>16MCA53X</td>
<td>Elective – 6</td>
<td>MCA</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>16MCA54X</td>
<td>Elective – 7</td>
<td>MCA</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>16MCA55</td>
<td>Minor Project II</td>
<td>MCA</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>16</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>16</td>
<td>0</td>
</tr>
</tbody>
</table>

**Elective 5**

(5 Credits) Theory + Practice

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Title</th>
<th>Course Code</th>
<th>Course Title</th>
<th>Course Code</th>
<th>Course Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>16MCA521</td>
<td>Cloud Computing</td>
<td>16MCA522</td>
<td>Virtual Reality</td>
<td>16MCA523</td>
<td>Internet of Things</td>
</tr>
</tbody>
</table>

**Elective 6**

(5 Credits)

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Title</th>
<th>Course Code</th>
<th>Course Title</th>
<th>Course Code</th>
<th>Course Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>16MCA531</td>
<td>Distributed and Parallel Computing</td>
<td>16MCA532</td>
<td>Service Oriented Architecture</td>
<td>16MCA533</td>
<td>Data warehousing &amp; Data Mining</td>
</tr>
</tbody>
</table>

**Elective 7**

(5 Credits)

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Title</th>
<th>Course Code</th>
<th>Course Title</th>
<th>Course Code</th>
<th>Course Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>16MCA541</td>
<td>Wireless and Mobile Networks</td>
<td>16MCA542</td>
<td>Principles of UI / UX Design</td>
<td>16MCA543</td>
<td>Soft Computing</td>
</tr>
</tbody>
</table>
### SIXTH SEMESTER

<table>
<thead>
<tr>
<th>Sl. No</th>
<th>Course Code</th>
<th>Course Title</th>
<th>BoS</th>
<th>CREDIT ALLOCATION</th>
<th>Total Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Lecture L</td>
<td>Tutorial T</td>
</tr>
<tr>
<td>1</td>
<td>16MCA61</td>
<td>Major Project</td>
<td>MCA</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>2</td>
<td>16MCA62</td>
<td>Seminar</td>
<td>MCA</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td></td>
<td></td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**Total Number of Credits Required to be Earned by Students**

<table>
<thead>
<tr>
<th>Program</th>
<th>Normal Duration Years (semesters)</th>
<th>Total No. of credits to be Earned (Average/Semester=25)</th>
</tr>
</thead>
<tbody>
<tr>
<td>PG Degree MCA</td>
<td>Three(3) Years or six semesters</td>
<td>150</td>
</tr>
<tr>
<td>PG Degree MCA. (Lateral Entry)</td>
<td>Two years (2) or four semesters</td>
<td>100</td>
</tr>
</tbody>
</table>

**Credit Distribution in the MCA Program**

<table>
<thead>
<tr>
<th>Category</th>
<th>Min Credits</th>
<th>Max Credits</th>
<th>Credits to be earned for 2016 Scheme</th>
</tr>
</thead>
<tbody>
<tr>
<td>Core (incl.soft core)</td>
<td>40</td>
<td>70</td>
<td>82</td>
</tr>
<tr>
<td>Elective</td>
<td>20</td>
<td>35</td>
<td>33</td>
</tr>
<tr>
<td>Seminar</td>
<td>02</td>
<td>10</td>
<td>02</td>
</tr>
<tr>
<td>Industrial Internship &amp; Project Work</td>
<td>20</td>
<td>35</td>
<td>33</td>
</tr>
<tr>
<td><strong>TOTAL</strong></td>
<td></td>
<td></td>
<td><strong>150</strong></td>
</tr>
</tbody>
</table>
# I SEMESTER

## DATA STRUCTURES
(Theory & Practice)

<table>
<thead>
<tr>
<th>Course Code</th>
<th>CIE Marks</th>
<th>Hrs/Week</th>
<th>SEE Marks</th>
<th>SEE Duration</th>
</tr>
</thead>
<tbody>
<tr>
<td>16MCA11</td>
<td>16MCA11</td>
<td>100+50</td>
<td>100+50</td>
<td>3 Hrs</td>
</tr>
</tbody>
</table>

### Credits: 5

### Course Learning Objectives (CLO)

Graduates shall be able to

1. Understand the fundamental techniques of Abstract Data Types
2. Implement different data structures like stacks, queues, linked lists, trees and graphs
3. Recognize different data structures and its applications
4. Solve problems by using data structures for different applications

### Unit – I | 10 Hrs

**Introduction to Data Structures**

Data Structures and Arrays in C, Implementing Structures and Union, Pointers, Scope of Variables Pointers and Dynamic Memory Allocation, Algorithm Specification, Data Abstraction

### Unit – II | 10 Hrs

**Arrays, Structures and Stacks**

Arrays, Dynamically Allocated Arrays, Structures and Unions, Sparse Matrices, Representation of Multidimensional Arrays, Stacks, Stacks using Dynamic Arrays, Evaluation of Expressions, Multiple Stacks

### Unit – III | 09 Hrs

**Queues and Linked lists**

Queues, Circular Queues, Single- and Double-Ended Priority Queues, Singly Linked lists and Chains, Representing Chains in C, Linked Stacks and Queues, Doubly Linked Lists

### Unit – IV | 09 Hrs

**Trees**

Binary Trees, Binary Tree Traversals, Threaded Binary Trees, Heaps, Binary Search Trees, Selection Trees, Forests, Counting Binary Trees

### Unit – V | 10 Hrs

**Graphs and Sorting**

The Graph Abstract Data Type, Graphs: Definitions, Applications of graphs, Representation of graphs, Bubble sort, Selection Sort, Merge sort, Tree sorting: Binary Tree sort, Heap Sort
Unit – VI (Lab Component)

Implement the following programs using C Language.

Part – A
1. Implement a menu driven program to search using -
   a. Linear Search  b. Binary Search
2. Write a menu driven program to sort the given number of elements (using random number generation) using
   a. Bubble Sort  b. Selection sort
3. Write a program to implement operations for a String based Stack
4. Write a program to implement basic queue operations
5. Simulate the working of circular queue providing the following operations
   a. Insert  b. Delete  c. Display
6. Simulate the working of a singly linked list providing the following operations
   a. Insert at the beginning
   b. Insert at the end
   c. Insert at the position
   d. Display
7. Simulate working of a singly circular linked list providing the following operations
   a. Delete from the beginning
   b. Delete every alternate element
   c. Display and Insert is Mandatory
8. Create a binary search tree and implement tree traversal

Part – B
1. Demonstrate a program to implement Recursion
2. Parse Infix arithmetic expressions to postfix arithmetic expressions.
3. Demonstrate how a computer system evaluates an Expression.
4. Demonstrate a program to print the jobs waiting in a queue assigning priority to the jobs.
5. Write a program to generate the rank list of a student using dynamic memory allocation.
6. Simulate an undo operation using doubly linked list.
7. Perform Depth Wise Traversal of a graph.

Note: Students are required to implement all the programs in Part-A and Part B
Expected Course Outcomes
After going through this course the student will be able to:

**CO1**: Understand data abstraction and data structures such as stacks, queues, lists, trees and graphs
**CO2**: Identify relevant data structures to develop solutions for a particular problem
**CO3**: Examine the use of data structures in relevant applications
**CO4**: Evaluate different data structures to solve a real world problem

Reference Books:


Scheme of Continuous Internal Evaluation (CIE) for Theory
CIE will consist of Two Tests, Two Quizzes and Two assignments. The test will be for 30 marks each, quiz and assignment for 10 marks each. The total marks for CIE (Theory) will be 100 marks.

Scheme of Continuous Internal Evaluation (CIE) for Practical
CIE for the practical will be based on the performance of the student in the laboratory every week for 10 marks for every experiment. Finally, the weekly evaluated marks will be consolidated for 40 marks. One test will be conducted at the end of the semester for 10 marks. The total marks for CIE (Practical) will be for 50 marks. One question from Part A and one from Part B need to be executed. Change of program is not permitted.

Scheme of Semester End Evaluation (SEE) for Theory
The question paper will be for 100 marks and shall consist of 10 questions from five units with 20 marks each. Out of the 10 questions students have to answer five questions from each unit. The questions will have Internal Choice with maximum 3 sub divisions. Both the questions shall be of the same complexity in terms of COs and Bloom’s taxonomy level.

Scheme of Semester End Evaluation (SEE) for Practical
SEE for the practical will be based on writing proper program, execution and proper results for 40 marks and 10 marks for viva-voce. The total marks for SEE (Practical) will be 50 marks. Part A weightage will be 70% and Part B weightage will be 30% of 40 marks. One question from Part A and one from Part B need to be executed. Change of program is not permitted.

Mapping of COs with POs

<table>
<thead>
<tr>
<th>CO/PO</th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
<th>PO7</th>
<th>PO8</th>
<th>PO9</th>
<th>PO10</th>
<th>PO11</th>
<th>PO12</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>H</td>
<td>M</td>
<td>M</td>
<td>L</td>
<td>M</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>CO2</td>
<td>H</td>
<td>M</td>
<td>M</td>
<td>L</td>
<td>M</td>
<td>-</td>
<td>-</td>
<td>L</td>
<td>L</td>
<td>M</td>
<td>L</td>
<td>L</td>
</tr>
<tr>
<td>CO3</td>
<td>H</td>
<td>M</td>
<td>H</td>
<td>H</td>
<td>H</td>
<td>-</td>
<td>L</td>
<td>L</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>CO4</td>
<td>M</td>
<td>H</td>
<td>H</td>
<td>H</td>
<td>H</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

H-High, M-Medium, L-Low
### OPERATING SYSTEMS
(Theory & Practice)

<table>
<thead>
<tr>
<th>Course Code</th>
<th>CIE Marks</th>
<th>Hrs/Week</th>
<th>SEE Marks</th>
<th>SEE Duration</th>
</tr>
</thead>
<tbody>
<tr>
<td>16MCA12</td>
<td>100+50</td>
<td>L:T:P:S 4:0:2:0</td>
<td>100+50</td>
<td>3 Hrs</td>
</tr>
<tr>
<td>Credits</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### Course Learning Objectives (CLO)
Graduates shall be able to
1. Identify the concepts, principles and services of operating system
2. Understand the operating system functionalities managing with hardware
3. Analyze the structure and design decisions involved in the implementation of an operating system
4. Evaluate different algorithms related to different operating system components
5. Explore various operating system utility commands to manage operating system
6. Implement various operating system algorithms and its evaluation

#### Unit – I
**10 Hrs**

**Introduction to Operating Systems**
Operating system objectives and functions, evolution of operating systems, Unix – modular structure, kernel components, Unix commands – file, directory, process, simple filters.

#### Unit – II
**10 Hrs**

**Process Management**
Process, Process States, Process Description, Process Control, CPU Scheduler and Scheduling Algorithms

#### Unit – III
**10 Hrs**

**Concurrency Control**
Principles of Concurrency, Semaphore, Message Passing, Principles of Deadlock, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection and Dining Philosopher’s problem using semaphores

#### Unit – IV
**10 Hrs**

**Memory Management**
Swapping, Contiguous Memory Allocation, Paging, Segmentation, Demand Paging, Page Replacement and Allocation of Frames

#### Unit – V
**08 Hrs**

**File and Disk Management**
File Sharing, Protection, Directory Implementation, Allocation Methods, Free Space Management, Disk Structure, Disk Scheduling and Disk Management
Unit – VI (Lab Component)

Part – A

1. a) Create a file under a three level file hierarchy structure and change file into read only file and display the username, size of the file and modification date

b) Convert the last or first 4 lines of a file into uppercase and store in another file

c) Display the row in the calendar which contains the date in which a specified file was created and convert the date value to *

2. a) Display corresponding home directory of a login name or current login

b) Display the users in the current working directory along with user and group identifiers

c) Display all the System information – operating system, kernel etc

3. Write a C program to mimic – grep command to search particular pattern, and its occurrence

4. Write a C program to mimic – cp command

5. Given the list of processes, their CPU burst times, arrival times and priority, compute and display the average waiting time and average turnaround time using Priority Scheduling.

6. Write a C program to simulate the MFT (Multiprogramming with Fixed number of tasks) memory management technique.

7. Write a C program to implement FCFS disk scheduling algorithm.

Part – B

1. Write a shell script to display the current user/any other user details with CPU and Memory utilization.

2. Write a shell script to implement Secured Terminal Login.

3. Write a script to search for file details in a directory (including subdirectory) which is having maximum and minimum memory size.

4. Write a Script to list the users who have logged in and logged out on a specified date and check for users currently logged in from the list.

Expected Course Outcomes

After going through this course the student will be able to:

CO1: Explain the fundamentals of operating system components and its functionalities

CO2: Analyze the basic operating system resources and its management techniques

CO3: Apply algorithms to handle the operations of an operating system

CO4: Implement solutions for classical problems in managing the computer resources

Reference Books


Scheme of Continuous Internal Evaluation (CIE) for Theory
CIE will consist of Two Tests, Two Quizzes and Two assignments. The test will be for 30 marks each, quiz and assignment for 10 marks each. The total marks for CIE (Theory) will be 100 marks.

Scheme of Continuous Internal Evaluation (CIE) for Practical
CIE for the practical will be based on the performance of the student in the laboratory every week for 10 marks for every experiment.
Finally, the weekly evaluated marks will be consolidated for 40 marks. One test will be conducted at the end of the semester for 10 marks. The total marks for CIE (Practical) will be for 50 marks.
One question from Part A and one from Part B need to be executed.
Change of program is not permitted.

Scheme of Semester End Evaluation (SEE) for Theory
The question paper will be for 100 marks and shall consist of 10 questions from five units with 20 marks each. Out of the 10 questions students have to answer five questions from each unit. The questions will have Internal Choice with maximum 3 sub divisions. Both the questions shall be of the same complexity in terms of COs and Bloom’s taxonomy level.

Scheme of Semester End Evaluation (SEE) for Practical
SEE for the practical will be based on writing proper program, execution and proper results for 40 marks and 10 marks for viva-voce. The total marks for SEE (Practical) will be 50 marks.
Part A weightage will be 70% and Part B weightage will be 30% of 40 marks. One question from Part A and one from Part B need to be executed.
Change of program is not permitted.

CO-PO Mapping

<table>
<thead>
<tr>
<th>CO/PO</th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
<th>PO7</th>
<th>PO8</th>
<th>PO9</th>
<th>PO10</th>
<th>PO11</th>
<th>PO12</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>M</td>
<td>M</td>
<td>L</td>
<td>L</td>
<td>M</td>
<td>H</td>
<td>L</td>
<td>L</td>
<td>-</td>
<td>L</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>CO2</td>
<td>H</td>
<td>M</td>
<td>L</td>
<td>M</td>
<td>M</td>
<td>H</td>
<td>M</td>
<td>-</td>
<td>-</td>
<td>L</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>CO3</td>
<td>H</td>
<td>M</td>
<td>L</td>
<td>M</td>
<td>H</td>
<td>L</td>
<td>H</td>
<td>H</td>
<td>-</td>
<td>L</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>CO4</td>
<td>H</td>
<td>M</td>
<td>L</td>
<td>H</td>
<td>H</td>
<td>-</td>
<td>M</td>
<td>H</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

H-High, M-Medium, L-Low
WEB TECHNOLOGIES

Course Code : 16MCA13
CIE Marks : 100+50
Hrs/Week : L:T:P:S  3:0:2:2
SEE Marks : 100+50
Credits : 5
SEE Duration : 3 Hrs

Course Learning Objectives (CLO)
Graduates shall be able to
1. Explain the technologies used in web applications.
2. Demonstrate HTML5, JavaScript coding for web applications
3. Designing a creative websites using object based scripting concepts
4. Analyze dynamic HTML, XML integration with DOM

Unit – I
07 Hrs

Introduction to Web Technologies

Unit – II
07 Hrs

Front End Design

Unit – III
08 Hrs

Basics of JavaScript
Overview of JavaScript, Object orientation and JavaScript, Syntactic characteristics, Primitives, operations, and expressions, Screen output and keyboard input, Control statements, Object creation and modification, Arrays, Functions, Constructors, Pattern matching using regular expressions.

Unit – IV
08 Hrs

JavaScript and HTML5 Documents
The JavaScript Execution Environment, The Document Object Model, Elements Access in Java Script, Events and Event Handling, Handling Events from Body Elements, Handling Events from Text Box and password Elements, Dom Tree Traversal and Modification.

Unit – V
08 Hrs

Dynamic Documents with JavaScript
Introduction, Positioning Elements, Moving Elements, Element Visibility, Changing Colors and Fonts, Dynamic Content, Stacking Elements, Locating the Mouse Cursor, Reacting to a Mouse Click, Slow Movement of Elements.
JavaScript Objects: Introduction Math Object, String Object, Fundamentals of Characters and Strings, Methods of the String Object, Character-Processing Methods Searching Methods, Date Object, Boolean and Number Objects document Object
Unit – VI (Lab Component)

PART A

1. Create an HTML5 page to demonstrate the usage of
   a) Text Formatting tags,
   b) Links
   c) Images
   d) Tables
2. Create a web page with all types of Cascading style sheets.
3. Develop and demonstrate a HTML5 file that includes JavaScript script for the following
   problems:
      a) Input: A number n obtained using prompt
         Output: The first n Fibonacci numbers
      b) Input: A number n obtained using prompt
         Output: A table of numbers from 1 to n and their squares using alert
4. Develop and demonstrate, using JavaScript script, a HTML5 document that collects the USN (the valid format is: A digit from 1 to 4 followed by two upper-case characters followed by two digits followed by two upper-case characters followed by three digits; no embedded spaces allowed) of the user. Event handler must be included for the form element that collects this information to validate the input. Messages in the alert windows must be produced when errors are detected.
5. Design an XML document to store information about a student in an engineering college affiliated to VTU. The information must include USN, Name, Name of the College, Branch, Year of Joining, and e-mail id. Make up sample data for 3 students. Create a CSS style sheet and use it to display the document.

PART – B

1. Create WebPages using HTML5 and CSS for Employee Management Portal. The pages should have the following, but not limited to:
   a. Proper headings
   b. Links for more details
   c. Images where ever appropriate
   d. Provision to take feedback from the user
   a) Home Page
   b) The static home page must contain three frames
   c) Top frame: Logo and the college name and links to Home page, Login page, Registration page,
   d) Left frame: At least four links for navigation, which will display the catalog of respective links.
   e) For e.g.: When you click the link “MCA” the catalog for MCA
   f) Books should be displayed in the Right frame.
   g) Right frame: The pages to the links in the left frame must be loaded here. Initially this page contains description of the web site
3. Demonstrate a HTML5 and JavaScript functions for the following problems:
   a) Parameter: A string
      Output: The position in the string of the left-most vowel
   b) Parameter: A number
      Output: The number with its digits in the reverse order
4. Demonstrate Java Script for different dialog box options.
5. Demonstrate a login page using HTML5 and validate the username and password using JavaScript.

Self-Study Component
Topics on latest / emerging technologies will be assigned. Students are required to read white papers, publications, patents and prepare a report, give a seminar on the study undertaken. The self study will be reviewed and evaluated by a expert panel in two phases appointed by the Director, MCA.

Expected Course Outcomes
After going through this course the student will be able to:
CO1: Understand mark-up and scripting language concepts and their applications
CO2: Demonstrate the working of dynamic documents in web designing
CO3: Analyze appropriate content layout design and event handling techniques
CO4: Implement static web document using HTML5, CSS, JavaScript and XML

Reference Books

Scheme of Continuous Internal Evaluation (CIE) for Theory
CIE will consist of Two Tests, Two Quizzes and Two assignments. The test will be for 30 marks each, quiz and assignment for 10 marks each. The total marks for CIE (Theory) will be 100 marks.

Scheme of Continuous Internal Evaluation (CIE) for Practical
CIE for the practical will be based on the performance of the student in the laboratory every week for 10 marks for every experiment.
Finally, the weekly evaluated marks will be consolidated for 40 marks. One test will be conducted at the end of the semester for 10 marks. The total marks for CIE (Practical) will be for 50 marks.
One question from Part A and one from Part B need to be executed.
Change of program is not permitted.

Scheme of Semester End Evaluation (SEE) for Theory
The question paper will be for 100 marks and shall consist of 10 questions from five units with 20 marks each. Out of the 10 questions students have to answer five questions from each unit. The questions will have Internal Choice with maximum 3 sub divisions. Both the questions shall be of the same complexity in terms of COs and Bloom’s taxonomy level.

Scheme of Semester End Evaluation (SEE) for Practical
SEE for the practical will be based on writing proper program, execution and proper results for 40 marks and 10 marks for viva-voce. The total marks for SEE (Practical) will be 50 marks. Part A weightage will be 70% and Part B weightage will be 30% of 40 marks. One question from Part A and one from Part B need to be executed. Change of program is not permitted.

### Mapping of COs with POs

<table>
<thead>
<tr>
<th>CO/PO</th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
<th>PO7</th>
<th>PO8</th>
<th>PO9</th>
<th>PO10</th>
<th>PO11</th>
<th>PO12</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>H</td>
<td>H</td>
<td>M</td>
<td>H</td>
<td>H</td>
<td>M</td>
<td>M</td>
<td>M</td>
<td>-</td>
<td>L</td>
<td>L</td>
<td>M</td>
</tr>
<tr>
<td>CO2</td>
<td>H</td>
<td>H</td>
<td>M</td>
<td>M</td>
<td>H</td>
<td>-</td>
<td>M</td>
<td>M</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>L</td>
</tr>
<tr>
<td>CO3</td>
<td>H</td>
<td>M</td>
<td>H</td>
<td>M</td>
<td>H</td>
<td>L</td>
<td>L</td>
<td>L</td>
<td>-</td>
<td>L</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>CO4</td>
<td>H</td>
<td>M</td>
<td>H</td>
<td>M</td>
<td>H</td>
<td>M</td>
<td>M</td>
<td>M</td>
<td>L</td>
<td>-</td>
<td>L</td>
<td>L</td>
</tr>
</tbody>
</table>

H-High, M-Medium, L-Low
COMPUTER ORGANIZATION AND ARCHITECTURE

Course Code : 16MCA14
CIE Marks : 100

Hrs/Week : L:T:P:S 4:0:0:4
SEE Marks : 100

Credits : 5
SEE Duration : 3 Hrs

Course Learning Objectives (CLO)
Graduates shall be able to
1. Describe various data representations and explain how arithmetic and logical operations are performed by computers
2. Explain the basic operation and relationship between the different components of computer.
3. Understand the advanced architecture of microprocessors
4. Write assembly programs for 8086 microprocessors

Unit – I
09 Hrs

Number Systems and Boolean Algebra
Number systems, Logic gates: The AND Gate, The OR gate, the inverter and Buffer, The NAND gate, the NOR Gate, the exclusive OR gate, The Exclusive NOR Gates, The NAND Gate as a universal Gate, Gates with More than two inputs, Using Inverters to convert gates. Axiomatic Definition of Boolean Algebra, Basic Theorems and Properties of Boolean Algebra, Boolean Functions, Canonical and Standard Forms, Other Logic Operations.

Unit – II
10 Hrs

Combinational Logic and Sequential Logic

Unit – III
09 Hrs

Basic Structure of Computer and Machine Instructions
Computer Types, Functional Units, Basic Operational Concepts, Bus structures, Performance, Memory Location and Addresses Machine Instruction and Programmers, Memory Operations, Instructions & Instruction Sequencing, Basic Input/output Operations. Introduction to Microprocessor based computer system.

Unit – IV
10 Hrs

8086 Architecture
Introduction, Architecture of 8086 Microprocessor, Pin functions, Minimum / Maximum mode of operation

Unit – V
10 Hrs

8086 Programming
8086 instruction sets, addressing modes, Assembler directives, Programming examples

Self-Study Component
Topics on latest / emerging technologies relevant to the course will be assigned. Students are required to read white papers, publications, patents and prepare a report, give a seminar on the study undertaken. The self study will be reviewed and evaluated by a expert panel in two phases appointed by the Director, MCA.
Expected Course Outcomes
After going through this course the student will be able to:
CO1: Understand the organization and architecture of a computer system.
CO2: Design and implement programs using assembly language.
CO3: Analyze the need for Logic circuits in digital system.
CO4: Create logic circuits for real time applications.

Reference Books


Scheme of Continuous Internal Evaluation (CIE)
CIE will consist of Two Tests, Two Quizzes and Self study. The test will be for 30 marks each, quiz and self study for 20 marks. The total marks for CIE (Theory) will be 100 marks.

Scheme of Semester End Evaluation (SEE)
The question paper will be for 100 marks and shall consist of 10 questions from five units with 20 marks each. Out of the 10 questions students have to answer five questions from each unit. The questions will have Internal Choice with maximum 3 sub divisions. Both the questions shall be of the same complexity in terms of COs and Bloom’s taxonomy level.

CO-PO Mapping

<table>
<thead>
<tr>
<th>CO/PO</th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
<th>PO7</th>
<th>PO8</th>
<th>PO9</th>
<th>PO10</th>
<th>PO11</th>
<th>PO12</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>H</td>
<td>H</td>
<td>L</td>
<td>H</td>
<td>L</td>
<td>-</td>
<td>M</td>
<td>M</td>
<td>-</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CO2</td>
<td>H</td>
<td>M</td>
<td>M</td>
<td>L</td>
<td>L</td>
<td>L</td>
<td>M</td>
<td>H</td>
<td>-</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CO3</td>
<td>H</td>
<td>H</td>
<td>L</td>
<td>M</td>
<td>-</td>
<td>M</td>
<td>M</td>
<td>-</td>
<td>L</td>
<td>M</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CO4</td>
<td>H</td>
<td>H</td>
<td>L</td>
<td>H</td>
<td>M</td>
<td>L</td>
<td>L</td>
<td>L</td>
<td>-</td>
<td>L</td>
<td>L</td>
<td></td>
</tr>
</tbody>
</table>

H-High, M-Medium, L-Low
## DISCRETE MATHEMATICS

<table>
<thead>
<tr>
<th>Course Code</th>
<th>16MAT15</th>
<th>CIE Marks</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hrs/Week</td>
<td>L:T:P:S 4:2:0:0</td>
<td>SEE Marks</td>
<td>100</td>
</tr>
<tr>
<td>Credits</td>
<td>5</td>
<td>SEE Duration</td>
<td>3 Hrs</td>
</tr>
</tbody>
</table>

### Course Learning Objectives (CLO)

Graduates shall be able to:

1. Identify and apply basic concepts of set theory, counting techniques and induction to perform computational operations.
2. Apply formal methods of symbolic logic and proof techniques used to solve traditional Computing problems.
3. Rationale mathematically about basic structures used in computer science.
4. Demonstrate the proper use of function notation and Identify the most appropriate model based both on technology and the context of the situation.
5. Use graph theoretic models and data structures to model and solve some basic problems in Informatics

### Unit – I

**10 Hrs**

**Set Theory and Fundamental Logic**


### Unit – II

**10 Hrs**

**Logic**

Basic Connectives and truth table of logic, Logic equivalence: The laws of Logic, Logical implications: Rules of Inference, Quantifiers, the use of quantifiers, Proofs of theorems.

### Unit – III

**09 Hrs**

**Relations**

Cartesian Products and relations, Computer recognition: zero-one matrices and directed graphs Properties of relations, Equivalence relations, Posets and Hassee diagrams.

### Unit – IV

**09 Hrs**

**Functions**

Functions: plain and one-to-one, onto functions, Stirling numbers of the second kind, function composition and inverse functions, special functions.

### Unit – V

**10 Hrs**

**Graphs**

Definition of graph, basic concepts in graph theory, vertex degree , Sub graphs, Complement and graph isomorphism, Euler trails and circuits, Hamilton paths and cycles, Planar graphs, Graph coloring and chromatic polynomials.
**Expected Course Outcomes**

After going through this course the student will be able to:

**CO1**: Demonstrate the understanding of the fundamental concepts of discrete mathematical structures

**CO2**: Apply the knowledge and skills obtained to investigate and solve a variety of mathematical foundation problems

**CO3**: Analyze mathematical concepts like sets, reasoning, relational algebra and graph theory to solve the problems and optimize the solution

**CO4**: Justify the overall mathematical knowledge gained to interpret and analyze the problems in various fields

**Reference Books**


**Tutorial Component**

Additional problems will be solved in all units during tutorial hour.

**Scheme of Continuous Internal Evaluation (CIE)**

CIE will consist of Two Tests, Two Quizzes and Two assignments. The test will be for 30 marks each, quiz and assignment for 10 marks each. The total marks for CIE (Theory) will be 100 marks.

**Scheme of Semester End Evaluation (SEE)**

The question paper will be for 100 marks and shall consist of 10 questions from five units with 20 marks each. Out of the 10 questions students have to answer five questions from each unit. The questions will have Internal Choice with maximum 3 sub divisions. Both the questions shall be of the same complexity in terms of COs and Bloom’s taxonomy level.

**Mapping of COs with POs**

<table>
<thead>
<tr>
<th>CO/PO</th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
<th>PO7</th>
<th>PO8</th>
<th>PO9</th>
<th>PO10</th>
<th>PO11</th>
<th>PO12</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>H</td>
<td>M</td>
<td>L</td>
<td>L</td>
<td>L</td>
<td>-</td>
<td>L</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>CO2</td>
<td>H</td>
<td>M</td>
<td>M</td>
<td>-</td>
<td>-</td>
<td>L</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>CO3</td>
<td>H</td>
<td>M</td>
<td>L</td>
<td>M</td>
<td>-</td>
<td>-</td>
<td>L</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>CO4</td>
<td>H</td>
<td>H</td>
<td>L</td>
<td>H</td>
<td>L</td>
<td>-</td>
<td>L</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

H-High, M-Medium, L-Low
# II SEMESTER

## DATABASE SYSTEMS  
(Theory & Practice)

<table>
<thead>
<tr>
<th>Course Code</th>
<th>16MCA21</th>
<th>CIE Marks</th>
<th>100+50</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hrs/Week</td>
<td>L:T:P:S 4:0:2:0</td>
<td>SEE Marks</td>
<td>100+50</td>
</tr>
<tr>
<td>Credits</td>
<td>5</td>
<td>SEE Duration</td>
<td>3 Hrs</td>
</tr>
</tbody>
</table>

### Course Learning Objectives (CLO)
Graduates shall be able to
1. Explain database concepts and structures and terms related to database design, transactions and management
2. Demonstrate data modeling, normalization and development of the database
3. Formulate SQL statements for data definition, modification and retrieval of data
4. Analyze how databases are affected by real-world transactions
5. Design and build a simple database system

#### Unit – I  
08 Hrs

### Basic Concepts
Introduction to data, information, databases, database management system; Characteristics of database approach, Actors on the Scene, Advantages of using DBMS approach, Classification of Database Applications, Data models, Schema and instances, Three schema architecture and independence, DBMS Environment, Client/Server Architectures of DBMS, E-R Model – E-R Diagrams

#### Unit – II  
08 Hrs

### Data Models and Basic SQL
Introduction to Data Models, Relational Model Concepts, Relational Model Constraints and Relational Database Schemas, Keys, Dealing with Constraint Violations, E-R to Relational Mapping, Advantages of SQL, Data Definition Language and Data Types

#### Unit – III  
09 Hrs

### Structured Query Language
Data Manipulation language, Data Control Language, Data Query Language and all related commands. Queries using Group by and Order by clause & Join, Operators, Aggregate Functions, Commit, Rollback, Save point. Views: Introduction

#### Unit – IV  
09 Hrs

### Database Design Theory and Normalization
Informal Design Guidelines for Relation Schemas, Functional Dependencies, Normal Forms Based on Primary Keys, General Definitions of Second and Third Normal Forms, Boyce-Codd Normal Form, Multi-valued Dependency and Fourth Normal Form, Join Dependencies and Fifth Normal Form

#### Unit – V  
10 Hrs

### Transaction Processing and Concurrency Control
Introduction-Properties of Transaction, Serializability, Concurrency Control, Locking Mechanisms, Two Phase Commit Protocol, Dead lock
Unit – VI (Lab Component)

Part – A

1. Consider the scenario of a hospital system. Patients are treated in a single ward by the doctors assigned to them. Usually each patient will be assigned a single doctor, but in rare cases they will have two. Healthcare assistants also attend to the patients; a number of these are associated with each ward. Initially the system will be concerned solely with drug treatment. Each patient is required to take a variety of drugs a certain number of times per day and for varying lengths of time. The system must record details concerning patient treatment and staff payment. Some staff are paid part time and doctors and care assistants work varying amounts of overtime at varying rates (subject to grade). The system will also need to track what treatments are required for which patients and when and it should be capable of calculating the cost of treatment per week for each patient.

   a. Identify super key, candidate keys, primary keys, Referential Integrity
   b. Explain the cardinality and participation between entities in the problem
   c. Create an ER diagram and the schema relationship for the above scenario
   Create the relations
   d. Design and execute queries for listing out
      The patients examined by a doctor
      Healthcare assistants of a ward
      Cost of treatment per week by a patient

2. Write the ER design and Create the relational data base of the Company with the below requirements and work out the queries Requirements (assume any required for the queries)
   The company is organized into DEPARTMENTs. Each department has a name, number and an employee who manages the department. We keep track of the start date of the department manager. Each department controls a number of PROJECTs. Each project has a name, number and is located at a single location. We store each EMPLOYEE’s social security number, address, salary, sex, and birthdate. Each employee works for one department but may work on several projects. We keep track of the number of hours per week that an employee currently works on each project. We also keep track of the direct supervisor of each employee. Each employee may have a number of DEPENDENTs. For each dependent, we keep track of their name, sex, birthdate, and relationship to employee.

Queries: Create all the relations based on the above scenario and do the following queries:

   a. Retrieve the names of all employees who do not have supervisors
   b. Retrieve the names of all employees whose surname is same as their supervisors
   c. Retrieve the name of each employee who has a dependent with the same first name as the employee
   d. Retrieve the name of each employee who works on all the projects controlled by department number 5
   e. Retrieve the names of employees who have no dependents
   f. For each project on which more than two employees work, retrieve the project number, project name, and the number of employees who work on that project
3. Notown Records has decided to store information about musicians who perform on its albums (as well as other company data) in a database. Each musician that records at Motown has an SSN, a name, an address, and a phone number. Poorly paid musicians often share the same address, and no address has more than one phone. Each instrument used in songs recorded at Notown has a unique identification number, a name (e.g., guitar, synthesizer, flute) and a musical key (e.g., C, B-flat, E-flat). Each album recorded on the Notown label has a unique identification number, a title, a copyright date, a format (e.g., CD or MC), and an album identifier. Each song recorded at Notown has a title and an author. Each musician may play several instruments, and a given instrument may be played by several musicians. Each album has a number of songs on it, but no song may appear on more than one album. Each song is performed by one or more musicians, and a musician may perform a number of songs. Each album has exactly one musician who acts as its producer. A musician may produce several albums, of course. **Design and develop a database for the above scenario and do the following Queries and ensure to grant permissions for specified users to view the contents (create views and grant permission over the view)**

   a. List out the musician names, songs he has played, the album in which it has occurred and the title
   b. Find out the album done by the producer of the album and plays guitar as well as flute and has produced no of songs greater than the average songs present
   c. List out how many musicians are from Bengaluru and what type of instruments they play
   d. Find the musicians who can play all the instruments present

4. Consider the following information about a university database: Professors have a PROFID, a name, an age, a rank, and a research specialty. Projects have a project number, a sponsor name (e.g., UGC/AICTE/…), a starting date, an ending date, and a budget. Graduate students have an USN, a name, an age, and a degree program (e.g., MCA/ MPhil/BE/ME ..). Each project is managed by one professor (known as the project’s principal investigator). Each project is worked on by one or more professors (known as the project’s co-investigators). Professors can manage and/or work on multiple projects. Each project is worked on by one or more graduate students (known as the project’s research assistants). When graduate students work on a project, a professor must supervise their work on the project. Graduate students can work on multiple projects, in which case they will have a (potentially different) supervisor for each one. Departments have a department number, a department name, and a main office. Departments have a professor (known as the director) who runs the department. Professors work in one or more departments, and for each department that they work in, a time percentage is associated with their job. Graduate students have one major department in which they are working on their degree. Each graduate student has another, more senior graduate student (known as a student advisor) who advises him or her on what courses to take.
Create all the relations based on the above scenario and do the following queries and ensure to grant permissions for specified users to view the contents (create views and grant permission over the view)

a. Retrieve the names of all professors who do not have an ongoing project of more than 1 lakhs
b. Retrieve the names of all graduate students along with their senior graduate student and the professors under whom they work for
c. List the professors and the sum of their total budgeted projects
d. Retrieve the names of project assistants who have more than two professors as supervisors and one of the supervisor is the director

**PART-B**

1. Create a GUI for each of the above scenarios and demonstrate CRUD operations
2. Backup and Restore Databases and tables
3. Using ODBC/JDBC, connect to the RDBMS and demonstrate CRUD operations

**Expected Course Outcomes**

After going through this course the student will be able to:

**CO1:** Explain the basic concepts of data models, database design for transaction processing and Query Language

**CO2:** Analyze an information storage problem and derive an information model expressed in the form of an entity relation diagram

**CO3:** Transform high-level conceptual model to relational data model, populate database and formulate queries based on principles of normalization

**CO4:** Design and Implement a Database for any given problem

**Reference Books**


**Scheme of Continuous Internal Evaluation (CIE) for Theory**

CIE will consist of Two Tests, Two Quizzes and Two assignments. The test will be for 30 marks each, quiz and assignment for 10 marks each. The total marks for CIE (Theory) will be 100 marks.

**Scheme of Continuous Internal Evaluation (CIE) for Practical**

CIE for the practical will be based on the performance of the student in the laboratory every week for 10 marks for every experiment.

Finally, the weekly evaluated marks will be consolidated for 40 marks. One test will be conducted at the end of the semester for 10 marks. The total marks for CIE (Practical) will be for 50 marks.

In test, the Part B can be executed for the data set created during execution of Part A.

Change of program is not permitted.
**Scheme of Semester End Evaluation (SEE) for Theory**

The question paper will be for 100 marks and shall consist of 10 questions from five units with 20 marks each. Out of the 10 questions students have to answer five questions from each unit. The questions will have Internal Choice with maximum 3 sub divisions. Both the questions shall be of the same complexity in terms of COs and Bloom’s taxonomy level.

**Scheme of Semester End Evaluation (SEE) for Practical**

SEE for the practical will be based on writing proper program, execution and proper results for 40 marks and 10 marks for viva-voce. The total marks for SEE (Practical) will be 50 marks.

Part A weightage will be 70% and Part B weightage will be 30% of 40 marks. One question from Part A and one from Part B need to be executed. The Part B can be executed for the data set created during execution of Part A.

Change of program is not permitted.

**Mapping of COs with POs**

<table>
<thead>
<tr>
<th>CO/PO</th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
<th>PO7</th>
<th>PO8</th>
<th>PO9</th>
<th>PO10</th>
<th>PO11</th>
<th>PO12</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>H</td>
<td>H</td>
<td>M</td>
<td>H</td>
<td>H</td>
<td>L</td>
<td>M</td>
<td>M</td>
<td>M</td>
<td>-</td>
<td>L</td>
<td>M</td>
</tr>
<tr>
<td>CO2</td>
<td>H</td>
<td>H</td>
<td>M</td>
<td>M</td>
<td>H</td>
<td>-</td>
<td>M</td>
<td>M</td>
<td>M</td>
<td>-</td>
<td>-</td>
<td>L</td>
</tr>
<tr>
<td>CO3</td>
<td>H</td>
<td>H</td>
<td>M</td>
<td>M</td>
<td>H</td>
<td>L</td>
<td>L</td>
<td>L</td>
<td>L</td>
<td>-</td>
<td>L</td>
<td>-</td>
</tr>
<tr>
<td>CO4</td>
<td>H</td>
<td>H</td>
<td>M</td>
<td>H</td>
<td>H</td>
<td>-</td>
<td>M</td>
<td>M</td>
<td>M</td>
<td>-</td>
<td>L</td>
<td>L</td>
</tr>
</tbody>
</table>

H-High, M-Medium, L-Low
### OBJECT ORIENTED PROGRAMMING

**(Theory & Practice)**

<table>
<thead>
<tr>
<th>Course Code</th>
<th>16MCA22</th>
<th>CIE Marks</th>
<th>100+50</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hrs/Week</td>
<td>L:T:P:S 4:0:2:0</td>
<td>SEE Marks</td>
<td>100+50</td>
</tr>
<tr>
<td>Credits</td>
<td>5</td>
<td>SEE Duration</td>
<td>3 Hrs</td>
</tr>
</tbody>
</table>

### Course Learning Objectives (CLO)

Graduates shall be able to

1. Develop an understanding of the essential principles in object oriented programming
2. Implement object oriented programming concepts using programming language
3. Incorporate design patterns standards for solving a real world problem
4. Utilize object based approaches during software development

### Unit – I

**10 Hrs**

#### Foundations of Object oriented Concepts


### Unit – II

**10 Hrs**

#### Design Principles


### Unit – III

**10 Hrs**

#### Design Patterns


### Unit – IV

**09 Hrs**

#### Introduction to Python Programming Language


### Unit – V

**09 Hrs**

#### Object oriented programming using Python

Classes and Objects, functions, methods. Inheritance, polymorphism and Exception handling
Unit – VI (Lab Component)

Part – A
1. Write a python script to demonstrate searching technique (linear or binary)
2. Write a python script to demonstrate sorting (Bubble or Selection or Insertion or Quick)
3. Write a python program demonstrating polymorphism (operator and function)
4. Write a python program to demonstrate Inheritance and exception handling
5. Write a python program demonstrating 10 operations using python datatypes (any one)
   a) String b) List c) Tuple d) Sets and e) Dictionary

Part – B
Students will be given problem statements to implement any of the following design patterns using Use Case Diagram and Class Diagrams
1. Facade (Structural Pattern)
2. Model View Control (Structural Pattern)
3. Command (Behavior Pattern)
4. State Pattern (Behavior Pattern)
5. Abstract Factory (Creation Pattern)
6. Singleton (Creation Pattern)

Expected Course Outcomes
After going through this course the student will be able to:
CO1: Exhibit their competence in object oriented programming
CO2: Demonstrate object oriented concepts using Python Programming Language
CO3: Implement design patterns using Python
CO4: Produce and/or debug programs that illustrate principles of object oriented software Design and development

Reference Books

Scheme of Continuous Internal Evaluation (CIE) for Theory
CIE will consist of Two Tests, Two Quizzes and Two assignments. The test will be for 30 marks each, quiz and assignment for 10 marks each. The total marks for CIE (Theory) will be 100 marks.

Scheme of Continuous Internal Evaluation (CIE) for Practical
CIE for the practical will be based on the performance of the student in the laboratory every week for 10 marks for every experiment.
Finally, the weekly evaluated marks will be consolidated for 40 marks. One test will be conducted at the end of the semester for 10 marks. The total marks for CIE (Practical) will be for 50 marks. One question from Part A and one from Part B need to be executed.
Change of program is not permitted.
Scheme of Semester End Evaluation (SEE) for Theory
The question paper will be for 100 marks and shall consist of 10 questions from five units with 20 marks each. Out of the 10 questions students have to answer five questions from each unit. The questions will have Internal Choice with maximum 3 sub divisions. Both the questions shall be of the same complexity in terms of COs and Bloom’s taxonomy level.

Scheme of Semester End Evaluation (SEE) for Practical
SEE for the practical will be based on writing proper program, execution and proper results for 40 marks and 10 marks for viva-voce. The total marks for SEE (Practical) will be 50 marks.
Part A weightage will be 70% and Part B weightage will be 30% of 40 marks. One question from Part A and one from Part B need to be executed.
Change of program is not permitted.

<table>
<thead>
<tr>
<th>CO/PO</th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
<th>PO7</th>
<th>PO8</th>
<th>PO9</th>
<th>PO10</th>
<th>PO11</th>
<th>PO12</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>H</td>
<td>M</td>
<td>M</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>L</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>CO2</td>
<td>H</td>
<td>L</td>
<td>M</td>
<td>-</td>
<td>M</td>
<td>-</td>
<td>L</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>CO3</td>
<td>H</td>
<td>H</td>
<td>H</td>
<td>M</td>
<td>H</td>
<td>M</td>
<td>L</td>
<td>-</td>
<td>M</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>CO4</td>
<td>H</td>
<td>M</td>
<td>M</td>
<td>M</td>
<td>H</td>
<td>L</td>
<td>L</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

H-High, M-Medium, L-Low
**ANALYSIS AND DESIGN OF ALGORITHMS**  
*(Theory & Practice)*

<table>
<thead>
<tr>
<th>Course Code</th>
<th>16MCA23</th>
<th>CIE Marks</th>
<th>100+50</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hrs/Week</td>
<td>L:T:P:S 4:0:2:0</td>
<td>SEE Marks</td>
<td>100+50</td>
</tr>
<tr>
<td>Credits</td>
<td>5</td>
<td>SEE Duration</td>
<td>3 Hrs</td>
</tr>
</tbody>
</table>

**Course Learning Objectives (CLO)**  
Graduates shall be able to

1. Understand the need of different Algorithm techniques  
2. Apply mathematical preliminaries to the analysis and design stages of different types of algorithms.  
3. Analyze the algorithms based on time and space complexity  
4. Understand and develop a variety of techniques for designing algorithms both on uni- and Multi-processor technology.  
5. Develop new or re-use already existing efficient algorithms to solve problems.

<table>
<thead>
<tr>
<th>Unit – I</th>
<th>10 Hrs</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Introduction to Algorithms &amp; Divide and Conquer technique</strong></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Unit – II</th>
<th>10 Hrs</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Divide and Conquer &amp; Greedy Method</strong></td>
<td></td>
</tr>
</tbody>
</table>
Insertion Sort, Depth First Search and Breadth First Search, Topological Sorting, Knapsack Problem, Job Sequencing with Deadlines, Minimum-Cost Spanning Trees: Prim’s Algorithm, Kruskal’s Algorithm; Single Source Shortest Paths.

<table>
<thead>
<tr>
<th>Unit – III</th>
<th>10 Hrs</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Dynamic Programming &amp; Coping with Limitations of Algorithmic Power</strong></td>
<td></td>
</tr>
</tbody>
</table>
Warshall’s Algorithm, Floyd’s Algorithm for the All-Pairs Shortest Paths Problem, Single-Source, Shortest Paths: 0/1 Knapsack, The Traveling Salesperson problem.

<table>
<thead>
<tr>
<th>Unit – IV</th>
<th>08 Hrs</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Space and Time Trade Offs and Limitations of Algorithmic Power</strong></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Unit – V</th>
<th>10 Hrs</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Backtracking and Branch - Bound Technique</strong></td>
<td></td>
</tr>
</tbody>
</table>
Backtracking: n-Queens problem, Hamiltonian Circuit Problem, Subset – Sum Problem, Branch and Bound-Assignment Problem, Travelling Salesman Problem, Approximation Approaches- Nearest Neighbor, Twice Around the Tree.

| Unit – VI (Lab Component) |
Design, develop and implement the specified algorithms for the following problems using C/C++. Students are required to execute all the programs in Part-A and part B, and show the demonstration in the lab.
Part – A

1. Implement Quicksort and analyze its time complexity using different values of n (n is the number of inputs) and represent the complexity in a graph sheet. The input should be generated randomly.
2. Print all the nodes reachable from a given starting node in a digraph using BFS method.
3. Check whether a given graph is connected or not using DFS method.
4. Find Minimum Cost Spanning Tree of a given undirected graph using Prim’s algorithm.
5. From a given vertex in a weighted connected graph, find shortest paths to other vertices using Dijkstra's algorithm.
6. Find Minimum Cost Spanning Tree of a given undirected graph using Kruskal’s algorithm and determine the time taken to find the minimum cost.
7. Compute the transitive closure of a given directed graph using Warshall's algorithm.
8. Implement 0/1 Knapsack problem using Dynamic Programming.
9. Find a subset of a given set \( S = \{s_1, s_2, \ldots, s_n\} \) of \( n \) positive integers whose sum is equal to a given positive integer \( d \). For example, if \( S= \{1, 2, 5, 6, 8\} \) and \( d = 9 \) there are two solutions \( \{1,2,6\} \) and \( \{1,8\} \). A suitable message is to be displayed if the given problem instance doesn't have a solution.
10. Implement N Queen's problem using Back Tracking.

Part – B

1. The time complexity of bubble sort is \( O(n) \). Suggest the improvements to be made in the algorithm so that the efficiency of the algorithm is improved.
2. Apply Divide and Conquer method to sort a given set of elements using MergeSort and determine the time required to sort the elements. The elements can be read from a file or can be generated using the random number generator.
3. Implement Dynamic programming to find solution to Traveling Salesperson problem and then solve the same problem instance using any approximation algorithm. Determine the error in the approximation.
4. Apply dynamic Programming to find the Shortest Path in a network among all the nodes.
5. Apply Decrease and Conquer Technique to topological order the vertices in a given digraph.

Expected Course Outcomes
After going through this course the student will be able to

CO1: Recognize and understand paradigms and approaches used to analyze and design algorithms and to appreciate the impact of algorithm design in practice.

CO2: Use different computational models (e.g., divide-and-conquer), order notation and various complexity measures (e.g., running time, disk space) to analyze the complexity/performance of different algorithms.

CO3: Implement and apply various techniques for efficient algorithm design (divide-and-conquer, greedy, and dynamic algorithms)

CO4: Compare different algorithm techniques for a real life application and find the optimal solution using various parameters
### Reference Books

<table>
<thead>
<tr>
<th></th>
<th>Book</th>
<th>Publisher</th>
<th>Edition</th>
<th>ISBN</th>
</tr>
</thead>
</table>

### Scheme of Continuous Internal Evaluation (CIE) for Theory

CIE will consist of Two Tests, Two Quizzes and Two assignments. The test will be for 30 marks each, quiz and assignment for 10 marks each. The total marks for CIE (Theory) will be 100 marks.

### Scheme of Continuous Internal Evaluation (CIE) for Practical

CIE for the practical will be based on the performance of the student in the laboratory every week for 10 marks for every experiment. Finally, the weekly evaluated marks will be consolidated for 40 marks. One test will be conducted at the end of the semester for 10 marks. The total marks for CIE (Practical) will be for 50 marks. One question from Part A and one from Part B need to be executed.

Change of program is not permitted.

### Scheme of Semester End Evaluation (SEE) for Theory

The question paper will be for 100 marks and shall consist of 10 questions from five units with 20 marks each. Out of the 10 questions students have to answer five questions from each unit. The questions will have Internal Choice with maximum 3 sub divisions. Both the questions shall be of the same complexity in terms of COs and Bloom’s taxonomy level.

### Scheme of Semester End Evaluation (SEE) for Practical

SEE for the practical will be based on writing proper program, execution and proper results for 40 marks and 10 marks for viva-voce. The total marks for SEE (Practical) will be 50 marks. Part A weightage will be 70% and Part B weightage will be 30% of 40 marks. One question from Part A and one from Part B need to be executed.

Change of program is not permitted.

### Mapping of COs with POs

<table>
<thead>
<tr>
<th>CO/PO</th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
<th>PO7</th>
<th>PO8</th>
<th>PO9</th>
<th>PO10</th>
<th>PO11</th>
<th>PO12</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>H</td>
<td>M</td>
<td>L</td>
<td>-</td>
<td>L</td>
<td>-</td>
<td>L</td>
<td>M</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>CO2</td>
<td>H</td>
<td>H</td>
<td>M</td>
<td>M</td>
<td>H</td>
<td>-</td>
<td>-</td>
<td>M</td>
<td>-</td>
<td>-</td>
<td>L</td>
<td>L</td>
</tr>
<tr>
<td>CO3</td>
<td>H</td>
<td>L</td>
<td>L</td>
<td>L</td>
<td>-</td>
<td>-</td>
<td>L</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>L</td>
<td>-</td>
</tr>
<tr>
<td>CO4</td>
<td>M</td>
<td>M</td>
<td>H</td>
<td>M</td>
<td>-</td>
<td>L</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

H-High, M-Medium, L-Low
<table>
<thead>
<tr>
<th>SOFTWARE ENGINEERING</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Course Code</strong>: 16MCA24</td>
</tr>
<tr>
<td><strong>Hrs/Week</strong>: L:T:P:S 3:2:0:4</td>
</tr>
<tr>
<td><strong>Credits</strong>: 5</td>
</tr>
</tbody>
</table>

**Course Learning Objectives (CLO)**
Graduates shall be able to
1. To explain the basic terminologies and implement systems effectively using various system models
2. To comprehend the testing Process and software evolution in order to meet dynamic changing requirements
3. To develop understanding of advanced concepts and methods required for construction of large software systems
4. To apply project management strategies for effective software development

### Unit – I 08 Hrs
**Introduction & Software Process Models**

### Unit – II 07 Hrs
**System Modeling, Architectural Design and implementation**

### Unit – III 07 Hrs
**Software Testing & Evolution**
Software Testing: Development testing, Test driven development, Release testing, User testing; Software Evolution: Evolution processes, Legacy systems, Software maintenance

### Unit – IV 07 Hrs
**Advanced Software Engineering**
Component-based Software Engineering: Components and component models, CBSE processes, Component composition; Distributed Software Engineering: Distributed systems, Client–server computing, Software as a service

### Unit – V 07 Hrs
**Software Management**
Project management: Risk management, Managing people, Teamwork; Project planning: Plan driven development, Project scheduling, Estimation techniques

**Self Study Component**
Topics on latest / emerging technologies will be assigned. Students are required to read white papers, publications, patents and prepare a report, give a seminar on the study undertaken. The self study will be reviewed and evaluated by an expert panel in two phases appointed by the Director, MCA.
Tutorial Component

Students are required to make the team of 4 to 5 members. Each team has to do role play (For Eg: customer, Analyst, Developer, tester & Manager) and exhibit different stages of various software life cycle models

Expected Course Outcomes

After going through this course the student will be able to:
CO1: Understand the basic terminologies and various Process models associated with software Engineering
CO2: Comprehend the testing Process and software evolution
CO3: Analyze and Apply advanced software engineering concepts and methods for construction of large software systems
CO4: Evaluate project management strategies for effective software development

Reference Books


Scheme of Continuous Internal Evaluation (CIE) for Theory

CIE will consist of Two tests, two quizzes and self study. The test will be for 30 marks each, quiz for 10 marks each and self study for 20 marks. The total marks for CIE (Theory) will be 100 marks.

Scheme of Semester End Evaluation (SEE) for Theory

The question paper will be for 100 marks and shall consist of 10 questions from five units with 20 marks each. Out of the 10 questions students have to answer five questions by selecting one from each unit. The questions will have Internal Choice with maximum 3 sub divisions. Both the questions shall be of the same complexity in terms of COs and Bloom’s taxonomy level.

Mapping of COs with POs

<table>
<thead>
<tr>
<th>CO/PO</th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
<th>PO7</th>
<th>PO8</th>
<th>PO9</th>
<th>PO10</th>
<th>PO11</th>
<th>PO12</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>M</td>
<td>-</td>
<td>M</td>
<td>M</td>
<td>H</td>
<td>L</td>
<td>L</td>
<td>M</td>
<td>H</td>
<td>L</td>
<td>L</td>
<td>-</td>
</tr>
<tr>
<td>CO2</td>
<td>L</td>
<td>-</td>
<td>L</td>
<td>M</td>
<td>M</td>
<td>L</td>
<td>L</td>
<td>M</td>
<td>M</td>
<td>-</td>
<td>M</td>
<td>L</td>
</tr>
<tr>
<td>CO3</td>
<td>L</td>
<td>L</td>
<td>H</td>
<td>M</td>
<td>M</td>
<td>L</td>
<td>L</td>
<td>H</td>
<td>L</td>
<td>L</td>
<td>L</td>
<td></td>
</tr>
<tr>
<td>CO4</td>
<td>M</td>
<td>-</td>
<td>M</td>
<td>L</td>
<td>M</td>
<td>M</td>
<td>L</td>
<td>H</td>
<td>M</td>
<td>L</td>
<td>L</td>
<td>M</td>
</tr>
</tbody>
</table>

H-High, M-Medium, L-Low
MANAGEMENT INFORMATION SYSTEMS & E-COMMERCE

Course Code : 16MCA25
CIE Marks : 100
Hrs/Week : L:T:P:S 4:0:0:4
SEE Marks : 100
Credits : 5
SEE Duration : 3 Hrs

Course Learning Objectives (CLO)
Graduates shall be able to
1. Understand the basic working principles of information systems and enterprises
2. Equip the students with preliminaries of technologies used in business information systems
3. Familiarize students with the Business applications and eCommerce initiatives
4. Enable the students to build decision support systems
5. Enhance the knowledge of the student about the management Security challenges in IT sector

Unit – I
Introduction to Information Systems in Business

Unit – II
Computer Hardware and Software
Computer Hardware – Trends in Computer Systems, Storage Trends and Trade Offs; Computer Software – Software Suites and Integrated Packages, Programming Packages; Business Telecommunication – Networking the Enterprise, Managing Organizational Change, Global Business and IT Strategies, Business Use of Internet; Database Management - Managerial Considerations for Data Resource Management (2 Case studies)

Unit – III
Information Systems for Business, eCommerce and Enterprise Collaboration

Unit – IV
Information Systems for Decision Support, Strategic Advantages
Introduction, Decision Support Systems (DSS), Using DSS, Executive Information Systems; Competitive Strategy Concepts, Strategic roles of Information Systems, Challenges of Strategic Information systems, Sustaining strategic success (2 Case studies)

Unit – V
Management Security Challenges & Controls
Organization and Information Technology, Security and Ethical Challenges: Information systems controls, its need, Audit information systems, Ethical dimensions, Computer Crime, Societal solutions, you and ethical responsibility (2 Case studies)
Self-Study Component
Topics on latest / emerging technologies will be assigned. Students are required to read white papers, publications, patents and prepare a report, give a seminar on the study undertaken. The self study will be reviewed and evaluated by an expert panel in two phases appointed by the Director, MCA.

Expected Course Outcomes
After going through this course the student will be able to:
CO1: Understand the fundamentals of a computer based information systems and enterprises
CO2: Analyze the technologies associated with business information systems
CO3: Apply eCommerce initiatives in various Business applications
CO4: Evaluate significance of support systems in enterprises and align to security control measures in IT sector

Reference Books

<table>
<thead>
<tr>
<th></th>
<th>Title</th>
<th>Author(s)</th>
<th>Publisher</th>
<th>Edition</th>
<th>ISBNS</th>
</tr>
</thead>
</table>

Scheme of Continuous Internal Evaluation (CIE) for Theory
CIE will consist of Two tests, two Quizzes and Self study. The test will be for 30 marks each, quiz for 10 marks each and self study for 20 marks. The total marks for CIE (Theory) will be 100 marks

Scheme of Semester End Evaluation (SEE) for Theory
The question paper will be for 100 marks and shall consist of 10 questions from five units with 20 marks each. Out of the 10 questions students have to answer five questions by selecting one from each unit. The questions will have Internal Choice with maximum 3 sub divisions. Both the questions shall be of the same complexity in terms of COs and Bloom’s taxonomy level.

Mapping of COs with POs

<table>
<thead>
<tr>
<th>CO/PO</th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
<th>PO7</th>
<th>PO8</th>
<th>PO9</th>
<th>PO10</th>
<th>PO11</th>
<th>PO12</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>M</td>
<td>-</td>
<td>L</td>
<td>-</td>
<td>M</td>
<td>M</td>
<td>L</td>
<td>H</td>
<td>M</td>
<td>L</td>
<td>L</td>
<td>M</td>
</tr>
<tr>
<td>CO2</td>
<td>L</td>
<td>L</td>
<td>M</td>
<td>H</td>
<td>M</td>
<td>M</td>
<td>-</td>
<td>M</td>
<td>M</td>
<td>L</td>
<td>-</td>
<td>M</td>
</tr>
<tr>
<td>CO3</td>
<td>L</td>
<td>M</td>
<td>M</td>
<td>H</td>
<td>M</td>
<td>M</td>
<td>L</td>
<td>M</td>
<td>M</td>
<td>L</td>
<td>L</td>
<td>M</td>
</tr>
<tr>
<td>CO4</td>
<td>L</td>
<td>-</td>
<td>L</td>
<td>M</td>
<td>M</td>
<td>H</td>
<td>-</td>
<td>M</td>
<td>M</td>
<td>H</td>
<td>L</td>
<td>L</td>
</tr>
</tbody>
</table>

H-High, M-Medium, L-Low