MAHATMA GANDHI UNIVERSITY
KOTTAYAM, KERALA

MASTER OF COMPUTER APPLICATIONS
REGULAR [3 YEARS]

REGULATIONS & SCHEME
For
AFFILIATED COLLEGES
(From 2017 admission onwards)
1. **Eligibility Conditions**

**Qualifications**

i) A candidate seeking admission to MCA course must have

* A pass with not less than 50% marks in any recognized regular bachelor's Degree course of minimum three years duration in any discipline with Mathematics at 10+2 level.

OR

* A pass with not less than 50% marks in any recognized Regular Bachelors Degree course of minimum three years duration in any discipline with Mathematics/Statistics/Business Mathematics/ Business Statistics as one of the Subjects.

OR

* A pass with not less than 50% marks in BCA/BSc Computer Science/ BSc Information Technology/ B.Techdegree of a minimum three years duration from a recognized University.

ii) Subject to the regulation relating to prescribed minimum of the respective qualifying examination, the minimum marks of admission to the course of studies shall be a pass in the case of SC/ST candidates.

iii) Candidates belonging to Socially and Educationally Backward Classes (SEBC) referred to GO(P)208/66/Edn dated 2-5-96 and subsequent amendments to orders issued by the Government and University shall be given a relaxation of 3% marks in the prescribed minimum for admission.

iv) A relaxation of 5% marks from the prescribed minimum shall be allowed in the case of OEC Candidates.

v) A relaxation of 5% marks from the prescribed minimum shall be allowed in the case of physically handicapped persons.

vi) Candidates who have passed the qualifying examination in more than one chance in the subject (excluding languages) will have their percentage marks de-rated at the rate of 5% for every additional appearance for the purpose of ranking.

Candidates with such degrees awarded by the Mahatma Gandhi University or any other degree recognized as equivalent to degrees in(i)by the Mahatma Gandhi University also are eligible to apply.

Reservation of seats shall be as per rules prescribed in the relevant rules by the Directorate of Technical Education, Government of Kerala from time to time.
2. **Duration of the Course**

The course shall extend over a period of three academic years consisting of six semesters.

3. **Requirements of attendance and progress**

A candidate will be deemed to have completed the course of any semester only if:

a) **He/She has put in not less than 75% of attendance,** and
b) **His/Her progress and conduct have been satisfactory.**

4. **Procedure for completing the Course**

i. The academic year will be divided into two semesters, the odd semester normally commencing at the beginning of the academic year and even semester ending with the academic year.

ii. The Course work in the subjects of study of the odd semesters will ordinarily be conducted only in odd semesters and that of even semesters only in even semesters.

iii. A candidate may proceed to the course of study of any semester if and only if he has completed the course in the previous semester and has registered for the examination of the previous semester.

iv. A candidate who is required to repeat the course of any semester for want of attendance / progress or who desires to rejoin the semester after a period of discontinuance or who upon his own request is specially permitted to repeat the semester in order to improve his performance, may be permitted to join the semester for which he is eligible or permitted to join.

5. **Assessment**

i) The assessment will comprise of sessional assessment and university examination in certain subjects, and wholly sessional assessments in others, carrying marks as specified in the subject of study and scheme of assessment.

ii) A candidate shall be declared to have passed in any subject in full in any semester if he/she secures not less than 50% marks in sessional, not less than 40% marks in the University examination including project and viva and not less than 50% of the over all aggregate marks for the subject i.e., university examination marks and sessional marks of the subjects put together.

iii) A student may be given the option to improve the marks obtained in theory subjects of any semester (except the sixth semester) by canceling all the theory examinations of the semester. There will be no provision to improve the sessional marks of any semester unless he repeats the semester.

vi) University examinations will be conducted at the end of each semester for subjects offered during the semester.

v) Semester examinations will normally be conducted in October/November and in April/May of each year.

vi) All Sessional work shall be valued and marks awarded on the basis of day to day performance, periodic tests and assignments. The allocation of sessional marks for individual subjects shall be on the following basis.
The sessional marks allotted for attendance shall be awarded in direct proportion to the percentage of attendance secured by the candidate in the subject. **However, full sessional marks for attendance shall be awarded to those who are securing 80% of the attendance and above only.**

6. Normalization of Sessional Marks

For the MCA course, the maximum internal marks(awarded internally) and external marks(awarded by external examiner appointed by the university) for all theory/practical papers shall be 25 and 75 respectively, except for the following papers - MCA108, MCA 407, MCA508.

To enforce uniformity in the awarding of internal marks by all institutions, there is a need to stipulate rules for normalizing the marks so that the abnormal and unjust variations in sessional marks are controlled to a reasonable extent.

For MCA 108, MCA 407 and MCA 508, having only sessional assessment, the Head of the Institution should ensure that the class average does not exceed 80%. For the remaining papers the following normalization method shall be implemented by the University.

**Normalization Method**

The maximum percentage of internal marks of a candidate shall be limited to 40% above that of external marks secured by the candidate.

In the case of a candidate who fails to get the pass minimum or absent for external examination for a paper, the normalized internal marks shall be computed only when he/she gets through the new external examination and the internal marks will be computed as per the new external marks.

**Illustration**

<table>
<thead>
<tr>
<th>Internal</th>
<th>Maximum Marks - 25</th>
<th>Pass Minimum -12.5</th>
</tr>
</thead>
<tbody>
<tr>
<td>External</td>
<td>Maximum Marks - 75</td>
<td>Pass Minimum – 30</td>
</tr>
<tr>
<td></td>
<td>Maximum Marks - 25</td>
<td>Pass Minimum - 10</td>
</tr>
<tr>
<td>Overall</td>
<td>Maximum Marks - 100</td>
<td>Pass Minimum -50</td>
</tr>
</tbody>
</table>
The above shall be computed using software and the normalized internal marks in the last column shall be carried over to the mark list.

7. University Exam Question Paper Pattern

The pattern shall comprise of 2 parts: **PART A** (10x3=30 marks) and **PART B** (5x9=45 marks).

**Part A** shall have 30 marks, in which the student is expected to answer 10 short questions (3 marks each) out of 12 questions **evenly prepared from all the five modules**. These questions can consist of definitions, theoretical concepts, short illustrative examples, block schematics etc.

**Part B** shall have 2 questions from each module, out of which the student has to answer one from each module (9 marks). These can be descriptive type questions, derivations, problems or collection of 2 or more small questions in a topic. This offers 50% choice to the students, yet forces him to study all the five modules.

8. Passing requirements/classification of successful candidates

i) A candidate shall be declared to have passed in any subject if he/she satisfies clause 5(ii) above.

   a) If any candidate fails in want of either minimum marks for university examination or minimum marks for overall aggregate for any subject, he/she can appear for the supplementary examination at the ensuing chance only in the failed subjects alone.

   b) If any candidate fails in want of minimum marks for **sessional part** alone for any subject, he/she has to write supplementary examination for both the **sessional part and university examination in the ensuing chance only in the failed subjects alone till he gets a pass mark for that subject**. Sessional part of such candidates may be evaluated by the institution, considering

<table>
<thead>
<tr>
<th>Reg. No.</th>
<th>External</th>
<th>Percentage</th>
<th>Max. % of internal eligible</th>
<th>Internal awarded by college</th>
<th>Internal marks after normalization</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>40</td>
<td>53%</td>
<td>93</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>2</td>
<td>15(failed)</td>
<td>20%</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>30(Next appearance)</td>
<td>40%</td>
<td>80%</td>
<td>22</td>
<td>20(limited to 80%)</td>
</tr>
<tr>
<td>3</td>
<td>60</td>
<td>80%</td>
<td>100</td>
<td>21</td>
<td>21</td>
</tr>
<tr>
<td>4</td>
<td>Absent 60(Next appearance)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>80%</td>
<td></td>
<td>100</td>
<td>18</td>
<td>18</td>
</tr>
</tbody>
</table>
the marks for attendance already obtained, but new assessment should be done for seminar/assignment and tests along with the subsequent batch. The new sessional mark has to be forwarded to the university along with the sessional marks of subsequent batch.

c) If any candidate fails in MCA 108, MCA 407 & MCA 508, having only sessional assessment, he/she has to redo the work for that subject along with the subsequent batch.

ii) A candidate who successfully completes the course and satisfy all the passing requirements of the six semesters within six academic years of joining the course will be declared to have qualified for the degree. However, in exceptional cases with genuine and convincing reasons, it is the discretion of the syndicate of the University to effect changes in this regard.

iii) A candidate who qualifies for the degree and secures not less than 75% of the aggregate of total marks of all the six semesters in the first attempt in all the subjects shall be declared to have passed the MCA Degree examination in First Class with Distinction.

iv) A candidate who qualifies for the degree and secures not less than 60% of the aggregate of total marks of all the six semesters shall be declared to have passed the MCA Degree examination in First Class.

v) All other successful candidates shall be declared to have passed the MCA Degree examination in Second Class.

vi) Successful candidates who complete the examinations with Distinction shall be ranked on the basis of the aggregate of the total marks of all six semesters.

9. Revision of Regulations

The University may from time to time revise, amend or change the regulations, curriculum, scheme of examinations and syllabi. These changes unless specified otherwise will have effect from the beginning of the next semester following the notification by the University.
# SCHEME OF THE PROGRAMME

Coding Structure: T: Theory; P: Practicals; S: Seminar; D: Dissertation; V: VivaVoce

*Example: MCA101T : MCA (Course) 101 (Paper Code) T(Theory Paper)*

## SEMESTER I

<table>
<thead>
<tr>
<th>CourseNo.</th>
<th>Subject</th>
<th>No. of hours per week</th>
<th>Duration of Exam in hrs</th>
<th>Sessional Marks Max.</th>
<th>Sem.Exam. Marks Max</th>
<th>Total Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>MCA101T</td>
<td>Discrete Mathematics and Statistics</td>
<td>4 - 3</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>MCA102T</td>
<td>Fundamentals of Data Structures</td>
<td>4 - 3</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>MCA103T</td>
<td>Paradigms of Programming Languages</td>
<td>4 - 3</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>MCA104T</td>
<td>Digital Systems &amp; Computer Architecture</td>
<td>4 - 3</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>MCA105T</td>
<td>Problem Solving and Programming in C</td>
<td>4 - 3</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>MCA106P</td>
<td>C Practicals</td>
<td>- 4</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>MCA107P</td>
<td>Data Structures through C - Practicals</td>
<td>- 4</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>MCA108T</td>
<td>English for Professional Communication</td>
<td>2</td>
<td>50</td>
<td>0</td>
<td>50</td>
<td></td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td><strong>750</strong></td>
</tr>
</tbody>
</table>

## SEMESTER II

<table>
<thead>
<tr>
<th>Course No.</th>
<th>Subject</th>
<th>No. of hours per week</th>
<th>Duration of Exam in hrs</th>
<th>Sessional Marks Max.</th>
<th>Sem. Exam. Marks Max</th>
<th>Total Mark</th>
</tr>
</thead>
<tbody>
<tr>
<td>MCA201T</td>
<td>Optimization Techniques &amp; Numerical Methods</td>
<td>4 - 3</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>MCA202T</td>
<td>Operating Systems</td>
<td>4 - 3</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>MCA203T</td>
<td>Database Management Systems</td>
<td>4 - 3</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>MCA204T</td>
<td>Data Communications &amp; Networks</td>
<td>4 - 3</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>MCA205T</td>
<td>Web Technologies</td>
<td>4 - 3</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>MCA206P</td>
<td>DBMS Practicals</td>
<td>4 - 3</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>MCA207P</td>
<td>Web Technologies Practicals</td>
<td>4 - 3</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td><strong>700</strong></td>
</tr>
</tbody>
</table>
## SEMESTER III

<table>
<thead>
<tr>
<th>Course No.</th>
<th>Subject</th>
<th>No. of hours per week</th>
<th>Duration of Exam in hrs</th>
<th>Sessional Marks Max.</th>
<th>Sem. Exam. Marks Max</th>
<th>Total Mark</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Lect</td>
<td>Lab.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MCA301T</td>
<td>Principles of Management &amp; Accounting</td>
<td>4</td>
<td>-</td>
<td>3</td>
<td>25</td>
<td>75</td>
</tr>
<tr>
<td>MCA302T</td>
<td>Analysis &amp; Design of Algorithms</td>
<td>4</td>
<td>-</td>
<td>3</td>
<td>25</td>
<td>75</td>
</tr>
<tr>
<td>MCA303T</td>
<td>Object Oriented Programming through Java</td>
<td>4</td>
<td>-</td>
<td>3</td>
<td>25</td>
<td>75</td>
</tr>
<tr>
<td>MCA304T</td>
<td>Software Engineering &amp; Project Management</td>
<td>4</td>
<td>-</td>
<td>3</td>
<td>25</td>
<td>75</td>
</tr>
<tr>
<td>MCA305T</td>
<td>Object Oriented Analysis &amp; Design</td>
<td>4</td>
<td>-</td>
<td>3</td>
<td>25</td>
<td>75</td>
</tr>
<tr>
<td>MCA306P</td>
<td>PHP Programming Practicals</td>
<td>-</td>
<td>4</td>
<td>3</td>
<td>25</td>
<td>75</td>
</tr>
<tr>
<td>MCA307P</td>
<td>OOPS through Java Practicals</td>
<td>-</td>
<td>4</td>
<td>3</td>
<td>25</td>
<td>75</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td>20</td>
<td>8</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

## SEMESTER IV

<table>
<thead>
<tr>
<th>Course No.</th>
<th>Subject</th>
<th>No. of hours per week</th>
<th>Duration of Exam in hrs</th>
<th>Sessional Marks Max.</th>
<th>Sem. Exam. Marks Max</th>
<th>Total Mark</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Lect</td>
<td>Lab.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MCA401T</td>
<td>System Software</td>
<td>4</td>
<td>-</td>
<td>3</td>
<td>25</td>
<td>75</td>
</tr>
<tr>
<td>MCA402T</td>
<td>Data Mining</td>
<td>4</td>
<td>-</td>
<td>3</td>
<td>25</td>
<td>75</td>
</tr>
<tr>
<td>MCA403T</td>
<td>TCP/IP Protocols</td>
<td>4</td>
<td>-</td>
<td>3</td>
<td>25</td>
<td>75</td>
</tr>
<tr>
<td>MCA404T</td>
<td>Linux OS and Shell programming</td>
<td>4</td>
<td>-</td>
<td>3</td>
<td>25</td>
<td>75</td>
</tr>
<tr>
<td>MCA405E</td>
<td>Elective- I</td>
<td>4</td>
<td></td>
<td>3</td>
<td>25</td>
<td>75</td>
</tr>
<tr>
<td>MCA406P</td>
<td>Linux OS &amp; Shell programming Practicals</td>
<td>-</td>
<td>4</td>
<td>3</td>
<td>25</td>
<td>75</td>
</tr>
<tr>
<td>MCA407D</td>
<td>Mini Project-Application Development</td>
<td>-</td>
<td>4</td>
<td>3</td>
<td>100</td>
<td>0</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td>20</td>
<td>8</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### SEMESTER V

<table>
<thead>
<tr>
<th>Course No.</th>
<th>Subject</th>
<th>No. of hours per week</th>
<th>Duration of Exam in hrs</th>
<th>Sessional Marks Max</th>
<th>Sem. Exam. Marks Max</th>
<th>Total Mark</th>
</tr>
</thead>
<tbody>
<tr>
<td>MCA501T</td>
<td>User Interface Design</td>
<td>4 -</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>MCA502T</td>
<td>Knowledge Management &amp; Business Intelligence</td>
<td>4 -</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>MCA503T</td>
<td>Enterprise Resource Planning</td>
<td>4 -</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>MCA504T</td>
<td>Advanced Java Programming</td>
<td>4 -</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>MCA505E</td>
<td>Elective - II</td>
<td>4 -</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>MCA506P</td>
<td>Advanced Java Programming Practicals</td>
<td>- 4</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>MCA507P</td>
<td>Python Programming Practicals</td>
<td>- 4</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>MCA508S</td>
<td>Main Seminar – Current Trends</td>
<td>2</td>
<td></td>
<td>50</td>
<td>0</td>
<td>50</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td></td>
<td></td>
<td>22</td>
<td>8</td>
<td>750</td>
</tr>
</tbody>
</table>

### SEMESTER VI

<table>
<thead>
<tr>
<th>Course No.</th>
<th>Subject</th>
<th>No. of hours per week</th>
<th>Duration of Exam in hrs</th>
<th>Sessional Marks Max</th>
<th>Sem. Exam. Marks Max</th>
<th>Total Mark</th>
</tr>
</thead>
<tbody>
<tr>
<td>MCA601D</td>
<td>Project</td>
<td>- 28</td>
<td>-</td>
<td>150</td>
<td>150</td>
<td>300</td>
</tr>
<tr>
<td>MCA602V</td>
<td>Viva-voce</td>
<td>- -</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>100</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td>- 28</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>400</td>
</tr>
</tbody>
</table>

**Elective I – Semester 4**

1. Microprocessor and Embedded Systems (E41)
2. Big Data Analytics (E42)
3. Cloud and Grid Computing (E43)
4. Social Network Analysis (E44)
5. Cryptography and Computer Security (E45)
6. Soft Computing (E46)

**Elective II – Semester 5**

1. Ad-hoc & Sensor Networks (E51)
2. Multimedia Systems (E52)
3. Information Security & E-Commerce (E53)
4. Digital Image Processing (E54)
5. Distributed Computing (E55)
6. Computer Graphics with Open GL (E56)
MCA 101T  DISCRETE MATHEMATICS & STATISTICS

UNIT I

UNIT II

UNIT III
Recurrence Relations: Generating Function of Sequences, Calculating Coefficient of generating functions, Recurrence relations, Solving recurrence relation by substitution and Generating functions, The method of Characteristic roots, Solution of Inhomogeneous Recurrence Relation. Graph Theory: Representation of Graph, Spanning Trees, BFS, DFS, Kruskals Algorithm, Binary trees, Planar Graphs Graph Theory and Applications, Basic Concepts, Isomorphism and Sub graphs, Multi graphs and Euler circuits, Hamiltonian graphs, Chromatic Numbers

UNIT IV

UNIT V
Tests of hypothesis- parameter and statistic-sampling distribution – Estimation and testing of hypothesis-critical region and level of significance-Errors in testing of hypothesis-one tailed and two tailed tests-procedure for testing hypothesis- confidence interval-tests of significance of large and small samples-Student’s t distribution- Snedecor’s F distribution.

REFERENCES
- Discrete Mathematical Structures with Applications to CS; Tremblery, R.Manohar, TMH
- Discrete Mathematical for computer Scientists & Mathematicians , Molt, Kandel, Baker, PHI
- T.Veerarajan-Probability , Statistics and Random process(Third edition ,TMH)
- Sundarapandian - Probability, Statistics and Queueing theory, PHI
- Purna Chandra Biswal – Probability and Statistics , PHI
- Elements of Discrete Mathematics, C L Liu, D P Mohanaptra,TMH
- Discrete Mathematical Structures, Kolman, Busby, Ross, 6th ed., PHI, 2009
UNIT I

UNIT II
Queues: Array and linked representation and implementation of queues, Operations on Queue: Create, Add, Delete, Full and Empty, Circular queues, D-queues and Priority Queues. Linked list: Representation and Implementation of Singly Linked Lists, Two-way Header List, Traversing and Searching of Linked List, Overflow and Underflow, Insertion and deletion to/from Linked Lists, Insertion and deletion Algorithms, Doubly linked list, Linked List in Array, Polynomial representation and addition, Generalized linked list, Garbage Collection and Compaction.

UNIT III

UNIT IV
Sorting: Insertion Sort, Bubble Sorting, Quick Sort, Two Way Merge Sort, Heap Sort, Sorting on Different Keys, Practical consideration for Internal Sorting. Binary Search Trees: Binary Search Tree (BST), Insertion and Deletion in BST, Complexity of Search Algorithm, Path Length, AVL Trees, B-trees.

UNIT V

REFERENCES
- K Loudon, “Mastering Algorithms with C”, Shroff Publisher & Distributors Pvt. Ltd.
MCA 103T PARADIGMS OF PROGRAMMING LANGUAGES

UNIT I
Introduction - Language evaluation criteria, influences on language design, paradigms in programming, stages in translation;
Syntax Analysis - programming language syntax, regular expressions, finite automata, grammar types, derivation, grammar issues, recursive descent process, concrete and abstract syntax, LL grammar, LR grammar, programming the scanner and parser;

UNIT II
Imperative Programming Languages – Variables, Type Checking, Scope, Referencing Environments, Data Types, Arithmetic Expressions, Control Flow, Stack and Sub programs

UNIT III
OO programming Languages – C++ and its support for OOP; Java and OOPs, C# and OOPs, Ruby and OOPs, Python and OOPs, Event handling in Java, Programming in OO languages;
Functional Programming languages – LISP, SCHEME Language, Meta Language (ML), Haskell Language

UNIT IV
Logical Programming language – PROLOG, Extended Logic Programming,
Concurrent and Distributed Programming – FORK in UNIX, Threads in Java, Network Programming, Distributed Programming

UNIT V
Scripting Language- Pragmatics, key Concepts, Case Study: Python – values and Types, variables, Storage and Control, Bindings and Scope, Procedural Abstraction, Data Abstraction, Separate Compilation, Module Library;

REFERENCES:
1. Principles of Programing languages – A Paradigm Approach – Syed Buhari, McGraw Hill,
2. Concepts of Programming languages Robert W Sebesta 8/e, Pearson Education
UNIT I
Number systems and code. Number systems - Efficiency of number system, Decimal, Binary, Octal, Hexadecimal conversion from one to another- Binary addition, subtraction, multiplication and division, representation of signed numbers, addition and subtraction using 2’s complement and 1’s complement. Binary codes - BCD code, Excess 3 code, Gray code, Alpha-numeric code, Error detection codes, Error correcting code.

UNIT II

UNIT III
Computer Organisation: Basic structure of computers-Machine Instructions and programs: Memory Locations and addresses, Memory Operations, Instructions and Instruction sequencing, Addressing modes, Basic Input Output Operations, Subroutines. Central Processing Unit; Arithmetic & Logic Unit: Number Representation – Addition of positive numbers – Fast Adders – Signed Addition and Subtraction – Multiplication of positive numbers – Multiplication using Booth's algorithm - Fast Multiplication – Floating point numbers and Operations. The Processing Unit - Basic Concepts - Instruction execution cycle - sequencing of control signals - hardwired control - microprogrammed control - control signals - microinstructions- microprogram sequencing - Branch address modification- Pre fetching of microinstructions

UNIT IV

UNIT V
Basic Parallel Processing Architecture - Flynn’s Classification - SISD, MISD, SIMD, MIMD structures - Pipelining – Basic Concepts of pipelining, Instruction Pipelining, Hazards, Vector processing & Vector processors - Loosely Coupled & Tightly Coupled Systems - Instruction-level parallelism: Concepts of instruction-level parallelism (ILP), Superscalar, Superpipelined and VLIW processor architectures - Comparison of RISC and CISC.

REFERENCES:
- Digital principles and Applications- Albert Paul Malvino, Donald P Leach, McGraw Hill.
UNIT I

Introduction to algorithm, flowchart, structured programming concept, programs – Compiler, Interpreter. Introduction to C Language: The C character set, identifiers and keywords, data types, how floats and doubles are stored, constants, variables and arrays, declarations, expressions, statements, Lvalues and Rvalues, type conversion, symbolic constants.

UNIT II

Operators and expressions: Arithmetic operators, unary operator, relational and logical operator, assignment operators, the conditional operator, type conversion, Library function Data input and output: Single character input, single character output, scanf, printf, puts gets functions, interactive programming. Control statement: Branching: if else statement, Looping, nested control structure, switch statement, break statement, continue statement, comma operator, goto statement.

UNIT III

Functions: Overview, function prototypes, passing arguments to a function, recursion. Program structure: Storage classes, automatic variables, external variables, static variables, multifile program. Arrays: Defining an array, passing array to functions, multidimensional arrays, strings: one dimensional character array, array of strings.

UNIT IV

Pointers: Fundamentals, void pointer, null pointer, passing pointers to a function, pointers and one dimensional arrays, dynamic memory allocation, operation on pointers, pointers and multidimensional arrays, array of pointers, pointer to an array, pointers and strings, structure pointer, pointers to function, pointers and variable length arguments list, passing functions to other functions. Structures and unions: Defining a structure, processing a structure, user defined data types, structure and pointers, passing structure to function, self-referential structures, and union.

UNIT V

Data files: Why files, opening and closing a data file, reading and writing a data file, processing a data file, unformatted data file, concept of binary file. Low level programming: Register variable, bitwise operations, bit fields. Additional features of C: Enumeration, Command line parameters, Macros, C Preprocessor.

REFERENCES

- Ansi C programming Bronson, Cengage learning, C2009
- Understanding Pointers in C- Yashavant Kanetkar – BPB publication, 2009
- Let us C - Yashavant Kanetkar – BPB publication C. 1997
- C by discovery –1 s Foster – Pearson C 2005
- Working with C - Yashavant Kanetkar – BPB publication, 2008
Section A (Min 30 programs)

- Implementation of the various Data Types in C.
- Demonstration of Data type conversion (Hint: Usage of type casting).
- Implementation of various Storage Types.
- Demonstration of for loop.
- Demonstration of do...while loop.
- Demonstration of while loop.
- Demonstration of nested if (Hint: Use logical operators).
- Demonstration of switch... case structure.
- Implementation of arrays.
- Implementation of multidimensional arrays (Hint: implement matrix operation).
- Implementation of functions (Hint: Demonstrate call by value, call by schemes, passing of arrays).
- Demonstration of various string operations (Hint: Usage of user defined functions only allowed). Demonstration of pointer operations.
- Demonstration of recursion (Hint: GCD, factorial, Fibonacci series).
- Demonstration of Debugging a C program.

Section B (Min 20 Programs)

- Implementation of structures (Hint: simple structure operations, array of structures).
- Implementation of Union.
- Implementation of pointers to structures and unions.
- Demonstration of dynamic allocation of memory (Hint: malloc, calloc, realloc, free).
- Demonstration of sorting techniques (Hint: selection sort, bubble sort).
- Demonstration of searching techniques (Hint: linear search, binary search).
- Demonstration of bitwise operations.
- Demonstration of macro processing.
- Demonstration of various file operations. (Hint: Text file)
- Implementation of character counting, line counting and word counting for a file.
- Program to find the lengthiest line in a text file.
Students are directed to develop a minimum of 40 Programs applying all Data Structure concepts studied in the Data Structures Theory paper.

- Programs to represent sparse matrix manipulation using arrays.
- Programs to represent Singly Linked List.
- Programs to represent Doubly Linked List.
- Programs to represent Circular Linked List.
- Polynomial addition using Arrays and Linked List.
- Programs to represent Stack operations using array and linked list
- Programs to represent Queue operations using array and linked list
- Programs for Conversion of infix to postfix.
- Programs for Evaluation of Expressions.
- Programs to represent Binary Tree Traversals.
- Programs to represent Searching procedures (Linear search, Binary search and Interpolation search)
- Programs to represent sorting procedures (Selection, Bubble, Insertion, Quick, Heap, Merge)
- Program to find the minimum cost spanning tree using Prim’s Method.
- Program to implement 8-Queens Problem.
Communication: Meaning and process of communication, importance of effective communication, barriers to communication. Types of communication, principles of communication, essentials of effective communication.

Media of Communication: Written, oral, face-to-face, visual, audio Visual, merits and demerits of written and oral communication..

Communication Skills: Developing communication skills, Listening, Speaking, Reading-Writing (Oral & Written). Body language; Utility of aids in Communication.

Spoken Skills: Preparing for oral presentation, conducting presentations, Debates, Seminar, Speeches, Lectures, Interviews, Telephonic Conversation, Negotiations; Group Discussions.

Written Skills: Preparing of bio-data, seminar, paper, bibliography and official correspondence, Mechanics of writing, Formal & Informal writings, letters, paragraphing, precise, report writing, technical reports, length of written reports, organizing reports, writing technical reports; Creative writing;

REFERENCES:
5. Scot Ober "Contemporary Business Communication", Wiley India.
UNIT I
Introduction and Overview of the OR Modeling Approach - The origin of OR, the nature of OR, the impact of OR, defining the problem and gathering, data, Formulating a mathematical model, deriving solutions from the model, testing the model, Preparing to apply the model, implementation.
Introduction to Linear Programming - Formulation of linear programming problem (LPP), examples, Graphical solution, the LP Model, Special cases of Graphical method, assumptions of Linear Programming (LP),

UNIT II
Solving LPP - the Simplex Method - The essence of the simplex method, setting up the simplex method, algebra of the simplex method, the simplex method in tabular form, special cases in the simplex method, tie breaking in the simplex method, adopting to other model forms (Two Phase method, Big-M method), post optimality analysis.
Duality Theory and Sensitivity Analysis - The essence of duality theory, economic interpretation of duality, primal dual relationship, adapting to other primal forms, the role of duality in sensitive analysis, the dual simplex method

UNIT III
Transportation and Assignment Problems - The transportation problem, a stream line simplex method for the transportation problem, the assignment problem, a special algorithm for the assignment problem ; PERT and CPM - Network representation, Critical path (CPM) computations and PERT networks. Game Theory - The formulation of two persons, zero sum games, solving simple games- a prototype example, games with mixed strategies, graphical solution procedure, solving by linear programming, extensions

UNIT IV
Inventory Control- Determination of EOQ, Components, Deterministic Continuous & Deterministic Periodic Review Models, Stochastic Continuous & Stochastic Periodic Review Models. Sequencing- Two men two machines, Three Men Two Machines

UNIT V

REFERENCES
1. Operation Research, Kanti Swaroop
2. Operation Research,V.K. Kapoor
3. Operation Research,Paneer Selvam,PHI
4. Operations Research, Hillier & Lieberman, TMH
UNIT I

Evolution of operating systems:- Serial processing, Batch Processing, multiprogramming. Types of operating systems-Batch Operating System, Multi programming-Time sharing, Real time, distributed operating systems. Operating Systems Structures:- Systems Components, Operating System Services, System Calls, System Programs, System Structures, Virtual Machines Processor Management:- Job and process concept, Operating system view of processes, process-state transition diagram, PCB; Threads, Operating system services. Process Scheduling:- Types of schedulers, scheduling and performance criteria, scheduling algorithms, multiple processor scheduling.

UNIT II

Inter process synchronization and communication- Concurrent Processes- need for inter process synchronization, critical section problem, mutual exclusion-mutual exclusion algorithms, semaphores-definition; busy wait implementation, monitors, inter process communication using messages. Deadlocks: - Definition – Deadlock characterization - Resource allocation graph, methods for handling deadlocks, deadlock prevention, deadlock avoidance-safe state-resource allocation graph algorithm, Banker’s algorithm, deadlock detection, recovery from deadlock.

UNIT III

Memory Management:- Preliminaries-address binding, dynamic linking and loading, Overlays. Logical versus physical address space, Swapping, Contiguous allocation Paging-principles of page allocation. Structure of page table- hardware support, multi-level paging, hierarchical paging, inverted page table, shared pages. Segmentation-principles of operation, hardware, implementation of segment table, protection and sharing, fragmentation, segmentation with paging.

UNIT IV


UNIT V

File Management:- File structure, File types, File access, File attributes and File operations. Directories-Flat directory systems, hierarchical directory systems. File system implementation- Allocation methods, contiguous allocation, linked allocation, indexed allocation.

Case Study - Desktop OS - Linux & Windows, Mobile OS – Android and iOS

REFERENCES:

3. Andrew S. Tanenbaum, “Modern Operating System, Prentice Hall India
4. Learning Android OS – Oreilly Publishers
5. Learning iOS – Oreilly Publishers
UNIT I
Need for a Database Management System - The file based system, Limitations of file based system, The Database Approach, The Logical DBMS Architecture, Three level architecture of DBMS or logical DBMS architecture, Mappings between levels and data independence, The need for three level architecture, Physical DBMS Architecture, DML Pre-compiler, DDL Compiler, File Manager, Database Manager, Query Processor, Database Administrator, Data files indices and Data Dictionary, Commercial Database Architecture, Data Models

UNIT II
The Relational Model - Domains, Attributes, Tuple and Relation, Super keys Candidate keys and Primary keys for the Relations, Relational Constraints, Domain Constraint, Key Constraint, Integrity Constraint, Update Operations and Dealing with Constraint Violations.

Relational Algebra - Basic Set Operation, Cartesian Product, Relational Operations, Entity Relationship (ER) Model, Entities, Attributes, Relationships, More about Entities and Relationships, Defining Relationship for College Database, E-R Diagram, Conversion of E-R Diagram to Relational Database

UNIT III
Relational Database Integrity- The Keys, Referential Integrity, Entity Integrity; Redundancy and Associated Problems- Single-Valued Dependencies - Single-Valued Normalization – 1NF, 2NF, 3NF, Boyce Codd Normal Form, Desirable Properties of Decomposition, Attribute Preservation, lossless-join Decomposition, Dependency Preservation, Lack of redundancy, Rules of Data Normalization, Eliminate Repeating Groups, Eliminate Redundant Data, Eliminate Columns Not Dependent on Key.

UNIT IV

UNIT V
ENHANCED DATABASE MODELS - Object Oriented Database- Limitations of Relational databases, The need of Object oriented databases, Complex Data Types, Structured Types and Inheritance in SQL, Object-Oriented versus Object-Relational; Database and XML- Structured Semi structure and unstructured data, XML hierarchical tree data model, Documents DTD and XML schema, XML Documents & Database, XML query and transformation, Storage of XML data, XML database applications

REFERENCES
• Atul Kahate, Introduction to Database Management Systems, Pearson Education India, June 2004
• Introduction to Database Management Systems, by Atul Kahate, Publisher: Pearson Education India, Release Date: June 2004
• Distributed databases Principles and systems, Steffano Ceri, Giuseppe Pelagatti. TMH
• Introduction to Object-Oriented Databases (Computer Systems Series) Won Kim
UNIT I

UNIT II

UNIT III

UNIT IV

Public key and Secret key, DNS, Resource Records, Name Service, E-mail-Architecture and services, TELNET, File Transfer Protocol (FTP), Simple Network Management Protocol (SNMP), Simple Mail Transfer Protocol (SMTP) Hyper Text Transfer Protocol (HTTP), World Wide Web (WWW).

UNIT V

REFERENCES:
5. William Stalings, ” Data and Computer communications”, Pearson Education.
6. Andrew S Tanenbaum ,“Computer Networks”, Prentice Hall.
UNIT 1
Introduction to web - connecting to internet, www, IP address, URL, DNS, Hosting.
Protocols governing the web - IP, TCP, HTTP, FTP, TELNET, Web development strategies-
Responsive design, Prototyping, Customization, Web applications-Architecture, Web
Application VS Distributed Application, Web project-Web Apps and Web Archives, Web
team-Roles and responsibilities.

UNIT II
Web Page Designing: HTML: HTML and SGML, DHTML, list, table, images, frames, forms
CSS: introduction, Creating style sheets, forms, color, classes, Layer tag, XML: features,
Structures in XML, DTD, XML schemes, presenting and using XML

UNIT III
Scripting: Java script: Introduction, documents, forms, statements, functions, objects; event
and event handling;
Introduction to AJAX- Requests, Response, Events
VB Script: Data types, User action, Functions, loops and conditions

UNIT IV
Server Side Programming:
Introduction to active server pages (ASP)-How does it work, procedures, ASP forms, cookies,
conditions, sessions, ASP.NET web pages, ASP.NET web forms, ASP.NET MVC model;
Java server pages (JSP)-JSP application design, tomcat server, JSP objects, declaring
variables, and methods, debugging, sharing data between JSP pages, Session, Application: data
base action, development of java beans in JSP
Introduction to COM/DCOM-Component Object Model, Interfaces, COM architecture,
DCOM, DCOM architecture, proxy and stub

UNIT V
The Ruby Language - OOP with Ruby - Text Processing and Scripting, Overview of Rails-
Document Requests- Processing Forms- Rails Application with Databases – Layouts. Ruby on
Rails Ajax.

REFERENCES
3. Ramesh Bangia, “Internet and Web Design”, New Age International
4. Bhave, “Programming with Java”, Pearson Education
Professional Paperback – Import, 16 Aug 2007, Cloves, Carneiro Jr, Hampton Catlin,
Jeffrey Hardy.
Introduction to MySQL Database

Installation of MySQL

1. Table Design- Using foreign key and Normalization
2. Practice SQL Data Definition Language (DDL) commands
   - Table creation and alteration (include integrity constraints such as primary key, referential integrity constraints, check, unique and null constraints both column and table level
   - Other database objects such as view, index, cluster, sequence, synonym etc.
3. Practice SQL Data Manipulation Language (DML) commands
   - Row insertion, deletion and updating
   - Retrieval of data
   - Simple select query
   - Select with where options (include all relational and logical operators)
   - Functions: Numeric, Data, Character, Conversion and Group functions with having clause.
   - Set operators
   - Sorting data
   - Sub query (returning single row, multiple rows, more than one column, correlated sub query)
   - Joining tables (single join, self join, outer join)
   - Data manipulations using date functions
   - User defined functions in a query
   - Hints in queries to optimize performance
   - Manage ODBC/JDBC connections
4. Practice Transaction Control Language (TCL) commands (Grant, revoke, commit and save point options)
5. Usage of triggers, functions and procedures using PL/SQL constructs
6. 10 PL/SQL Programs to be developed demonstrating PL/SQL Concepts

7. Development of a Databases using MySQL that could be used with one of the following application:
   i) Payroll Information
   ii) Student Information System
   iii) Library Information System etc.

8. Backup and restore databases;
9. Query optimization in a database
Every student is expected to generate and develop a minimum of 30 programs (simple and complex) for the final evaluation and prepare a record of the same. All programs to be developed in Linux OS Environment.

Design an HTML form implementing different input types
Usage of internal and external CSS along with HTML pages
Create HTML page with layout
Create an image gallery using CSS
Use CSS to style an HTML form
Implementation of tree structure in XML
Implementing XML DTD
Java Script -Form validation including text field, radio buttons, checkboxes, list box and other Controls
Date handling using java Script
JS event Handling -Creating event handler that respond to mouse and keyboard event:
Onload, onmouseover, onmouseout, onfocus, onblur, onsubmit, onresult, onclick,
Program to implement JS functions
Basic JSP programs
Sharing data between JSP pages
Using sessions and cookies as part of web applications (using JSP)
Sample web applications using JSP and MySQL
Introduction to Rails; Simple programs on Rails, Rails Application with Databases
MCA 301T  PRINCIPLES OF MANAGEMENT & ACCOUNTING

UNIT I
Basic Managerial Concepts, Levels of management, Managerial Skills, Concept of management Principles, nature and need of management, management functions, management thought – classical approach, scientific management, Fayol’s management, bureaucratic approach, systems approach, Contingency approach. Planning – Meaning, nature, structure, steps, effective planning, MBO, SWOT Analysis. Organizing – meaning, process, structure, formal and informal, types of organization, departmentation, delegation of authority.

UNIT II

UNIT III
Organizational behavior – Key elements, scope, models of OB, Individual behavior, personality, attitudes values and job satisfaction, Group behavior, team building- Types, process, roles.

UNIT IV

UNIT V

References
1. Principles of Management, R N Gupta, S.Chand & Company Ltd.
5. Accounting for Management, Srinivasan & Murugan, S.Chand & Company Ltd
6. Organisational Behavior, S.S Khanka, S.Chand & Company Ltd
UNIT I

UNIT II
Divide and conquer method – General method, Finding the maximum and minimum, mergesort, Quick sort, Selection sort, Strassen’s matrix multiplication.

UNIT III

UNIT IV


UNIT V

REFERENCES
- Fundamentals of computer algorithms- Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajeshekaran
- Fundamentals of algorithms – Gilles Brassard, Paul Bratley (PHI)
- Introduction to the design and analysis of algorithms – AnanyLevitin (Pearson)
- Computer algorithms – Introduction to design and analysis – Sara Baase, Allen VanGelder, (Pearson)
- Algorithm Design, Foundation, Analysis and Examples,Dr.Vijayakumar & Dr.Juby Mathew, Vimala Publications ,2016
UNIT I
Object Oriented Programming: Introduction to OOP’s Paradigm, Characteristics of OOP’s.  
History and Basics of Java: Java’s History and Creation, Java’s Magic: Byte-code, Its Features, Architecture of Java Virtual Machine, Importance of Java for the Internet, JDK, Java Editors (notepad++, eclipse, netbeans, jbuilder, bluej) Java Program Structure and Java’s Class Library, Java Data Types, Variables, and Operators, Operator Precedence. Scope of Variables, Control Structure. Array and String: Declaration and Definition, String Handling Using String Class and it functions and Wrapper classes.

UNIT II
Extending Classes and Inheritance: Fundamental of Inheritance, Using Existing Classes, Polymorphism, Multiple Levels of Inheritance, Super keyword, super-class constructor, Abstraction through Abstract Classes, Using Final Modifier. Packages & Interfaces: Understanding Packages, Defining a Package, Packaging up Your Classes, Adding Classes from a Package to Your Program, Understanding CLASSPATH, Standard Packages, Access Protection in Packages, Concept of Interface, Multiple Inheritance through Interfaces.

UNIT III
Exception Handling: The concept of Exceptions, Types of Exceptions, Dealing with Exceptions, Exception Objects, Try and catch blocks, Try Defining Your Own Exceptions.   

UNIT IV

UNIT V

REFERENCES:
1. Object Oriented Programming With Java, Balagurusami
2. The Complete Reference JAVA by Herbert Schildt, TMH Publication.
4. JAVA 2 UNLEASHED, Tech Media Publications.
5. JAVA 2(1.3) API Documentations
UNIT I


UNIT II


UNIT III

Selection of an appropriate project approach: Choosing Technologies, technical plan contents list, choice of process models, structure versus speed of delivery, The Waterfall model, The V-process model, the spiral model, Software prototyping, other ways of categorizing software prototypes. Controlling changes during prototyping, incremental delivery, dynamic systems development method, Extreme programming Managing iterative processes, selecting the most appropriate process model

UNIT IV

Software Effort Estimation: Problems with over and under estimates. The basis for Software estimating, Software effort estimation techniques, expert judgment, estimating by analogy, Albrecht function point analysis, function points Mark II, Object points, a procedural code – oriented approach, COCOMO: A Parametric Model.

UNIT V

Activity planning: The objectives of activity planning, When to plan, Project Schedules, Projects and activities, Sequencing and scheduling activities, Network planning models, Formulating a network model, Adding the time dimension, Risk Management: The nature of risk, types of risks, Managing Risk, Hazard Identification, Hazard Analysis, Risk planning and control, Evaluating risks to the schedule.

REFERENCES

MCA 305T  OBJECT ORIENTED ANALYSIS & DESIGN

UNIT-I

AN OVERVIEW OF OBJECT ORIENTED SYSTEMS DEVELOPMENT: Introduction, Two Orthogonal Views of the Software, Object Oriented Systems Development Methodology, Why an Object Orientation? WHY WE MODEL: The Importance of Modeling, Principles of Modeling, Object Oriented Modeling


UNIT-II

BASIC STRUCTURAL MODELING: Classes, Relationships, Common Mechanisms, and diagrams, class diagrams ADVANCED STRUCTURAL MODELING: Advanced classes, advanced relationships, Interfaces, Types and Roles, Packages, Object Diagrams

UNIT-III

BASIC BEHAVIORAL MODELING: Interactions, Interaction diagrams, Use cases, Use case diagrams, Activity Diagrams ADVANCED BEHAVIORAL MODELING: Events and signals, state machines, processes and Threads, time and space, state chart diagrams.

UNIT-IV (11 Lectures)

ARCHITECTURAL MODELING I: Component, Deployment, Component diagrams and Deployment diagrams ARCHITECTURAL MODELING II: Patterns and Frameworks, Collaborations, Systems and Models.

UNIT-V

CASE STUDY: Bank ATM Application, Railway Reservation System.

REFERENCES:

MCA 306P PHP PROGRAMMING PRACTICALS

Installing PHP, creating and working with variables, constants, PHPs Internal Data Types;

Math Operators, Assignment Operators, String Operators, Bitwise Operators, Comparison Operators, Logical Operators;

Using the IF Statement, else Statement, elseif, ternary operator, switch statement, while loops, do..while loops, foreach loop;

String functions, formatting text strings, arrays, handling arrays with Loops

Sorting Arrays; Array Operators; Multidimensional Arrays;

Functions in PHP, passing arrays to functions, passing by reference, returning arrays, introducing variable scope in PHP, Conditional Functions, Nesting functions, Variable Functions;

Setting Webpages to communicate with PHP; Handling text fields, text areas, check boxes, radio buttons, list boxes, password controls, image maps, buttons, File uploads;

PHP server variables, HTTP Headers

Obect oriented Programming – creating classes, objects, setting access, Constructors, Inheritance, Overloading, Autoloading Classes

Static Methods, Creating Abstract Classes, Interfaces, FINAL Keyword;

File handling – fopen, feof, fgets, closing a file, fgetc, f_get_contents, file_exists, filesize, fread, fscanf, fseek, copying files, fwrite, reading and writing binary files, appending to files

Working with Database – MySQL Database, Connecting to the database server and database, displaying the table data, updating databases, inserting new data into database, creating new tables, deleting records, sorting the data.

Sessions and cookies – Setting a cookie, reading a cookie, working with ftp, downloading files with ftp, uploading and deleting files with ftp, sending email, writing a hit counter using sessions;

Implement the above concepts using 50 programs in the Lab Cycle and a small project connecting a website developed in PHP with a database created using MySQL.

References:

- The complete reference PHP, McGraw Hill Education, Holzner;
- PHP Programming, Penn Wu,
- Object oriented PHP, Peter Lavin
- Modern PHP, Josh Lockhart
All students are expected to develop 50 Programs (simple and complex) which will demonstrate the theoretical concepts studied in the Java Theory paper of MCA.

- Programs to illustrate class, objects and constructors
- Programs to implement overloading, overriding, polymorphism etc
- Programs to implement the usage of packages
- Programs to create our own exception
- Programs for handling file operation
- Implement the concept of thread Programming
- Programs to implement Generic class and generic methods
- Applet Programs for passing parameters
- Applet Programs for running an audio file
- Programs for event-driven paradigm in Java
- Event driven Programs for Graphical Drawing Application
- Programs that uses Menu driven Application
- Programs to implement JDBC in GUI and Console Application
- Web page design using HTML and client side validation using Java-script
- Programs to implement session Handling and Cookies in Servlets and JSP
- Socket Programming to implement communications
- Develop a multi-threaded GUI application of your choice.
UNIT I

UNIT II
INTRODUCTION - System software and machine architecture - The Simplified Instructional Computer (SIC) Machine architecture - Data and instruction formats - addressing modes - Instruction sets - I/O and programming.

UNIT III

UNIT IV

UNIT V

SYSTEM SOFTWARE TOOLS- Text editors -Overview of the Editing Process -User Interface – Editor Structure. -Interactive debugging systems-Debugging functions and capabilities – Relationship with other parts of the system – User-Interface Criteria

REFERENCES
UNIT I
Introduction to Data mining & Date Warehouse
What is Data mining, Data mining -On What kinds of Data, Data mining Functionalities, Classification of Data mining Systems, Data Mining Task Primitives, Integration of Data mining systems, Major issues of Data mining, What is Data Warehouse, Multidimensional Data Model, A three-tier Data Warehousing Architecture.

UNIT II
Data Preprocessing and Mining Frequent Patterns
Data Preprocessing; Data Cleaning, Data Integration and Transformation, Data Reduction, Data discretization and concept hierarchy generation.
Association Rules
Basic Concepts, Efficient and Scalable Frequent Item set Mining Methods : Apriori Algorithm, Generating association Rules from Frequent Item sets, improving the Efficiency of Apriori. Mining Frequent item-sets without Candidate Generation.

UNIT III
Classification and Prediction
Introduction to Classification and Prediction, Issues Regarding Classification and Prediction
Classification by Decision Tree Induction: Decision Tree induction, Attribute Selection Measures,Tree Pruning, Bayesian Classification: Bayes' theorem, Naïve Bayesian Classification, Rule Based Algorithms: Using If - Then rules of Classification, Rule Extraction from a Decision Tree, Rule Induction Using a Sequential Covering algorithm, K-Nearest Neighbour Classifiers.
Prediction : Linear Regression, Nonlinear Regression, Other Regression-Based Methods

UNIT IV
Clustering

UNIT V
Applications and Trends in Data Mining
Data Mining Applications : Data Mining for Financial Data Analysis, Data Mining for the Retail Industry, Data Mining for the Telecommunication Industry, Data Mining for Biological Data Analysis, Data Mining in Other Scientific Applications, Data Mining for Intrusion Detection, Social Impacts of Data Mining, Trends in Data Mining.

References
- Data Mining Concepts and Techniques – Jiawei Han and Micheline Kamber, Second Edition, Elsevier, 2006
- Data Warehousing, Data Mining, & OLAP – Alex Berson, Stephen Smith, TMHill, 2004
- Data Warehousing, Sinha, Thomson Learning
MCA 403T  TCP/IP Protocols

UNIT I

UNIT II
Addressing, Classful Addressing, Classless Addressing, Special Addresses, NAT; Delivery of Packets, Forwarding based on Destination Address, Forwarding based on Labels, Structure of a router; ARP- Address Mapping, The ARP Protocol, ATMARP, ARP Package; ICMP – Messages, Debugging Tools, ICMP Package;
Mobile IP- Addressing, Agents, Three Phases, Inefficiency in Mobile IP;

UNIT III
Unicasting, Intra and Inter Domain Routing, Distance Vector Routing, RIP, Link State Routing, OSPF, Path Vector Routing, BGP; Multicasting – Introduction, Multicast Addresses, IGMP, Multicast routing, Routing Protocols – Multicast Link State Routing, Multicast Distance Vector,DVMRP, CBT, PIM; Transport Layer Services; Transport layer protocols, UDP, UDP Services, UDP Applications

UNIT IV
TCP Services, Segment, State Transition Diagrams, flow Control, Error Control, Congestion Control; Client Server paradigm, Peer to Peer paradigm; DNS, DNS Resolution, Messages, Types of records;

UNIT V
TELNET, SSH, FTP, TFTP, HTTP, SMTP; IPv6 Addressing, IPv6 Protocols;

REFERENCES
- TCP/IP Illustrated: The Protocols, Fall & Stevens;
- TCP/IP foundations, Andrew Blank, Wiley Publishers
MCA 404T LINUX OS AND SHELL PROGRAMMING

UNIT I

UNIT II
Basics of shell programming, various types of shell available in Linux, comparisons between various shells, shell programming in bash - Conditional and looping statements, Iterations, Command Substitution - expr command, arithmetic expansion, parameter passing and arguments, Shell variables, system shell variables, shell keywords, Creating Shell programs for automating system tasks.

UNIT III
Common administrative tasks, identifying administrative files configuration and log files, Role of system administrator, Managing user accounts-adding & deleting users, changing permissions and ownerships, Creating and managing groups, modifying group attributes, Temporary disabling of users accounts, creating and mounting file system.

UNIT IV
Checking and monitoring system performance - file security & Permissions, becoming super user using su. Getting system information with uname, host name, disk partitions & sizes,users, kernel. Installing and removing packages. Backup, restore and Compress utilities - tar, cpio, dump, rsync and restore utilities.

UNIT V
Communication in Linux - mesg, who- T, talk, write, wall, finger, chfn, ping, traceroute utilities, email facilities. Configuration of servers- Telnet, FTP, DHCP, NFS, SSH, Proxy Server(Squid), Web server (Apache), Samba. Daemons- init, crond, atd, xinetd, inetd, the services file. named, sshd, httpd.

REFERENCES
- Operating System - Linux, NIIT Press, PHI Publisher, 2006 Edition
- Red Hat Linux Bible, Cristopher Negus, Wiley Dreamtech India
- UNIX Shell Programming by Yeswant Kanetkar, BPB
- Linux Administration Handbook, Evi Nemeth, Garth Snyder, Trent KHein –Pearson Education.
- Beginning Linux Programming by Neil Mathew & Richard Stones, Wiley Dreamtech
LIST OF ELECTIVES FOR SEMESTER 4

1. E41 – MICROPROCESSORS AND EMBEDDED SYSTEMS
2. E42 – BIG DATA ANALYTICS
3. E43 – CLOUD AND GRID COMPUTING
4. E44 – SOCIAL NETWORK ANALYSIS
5. E45 – CRYPTOGRAPHY AND COMPUTER SECURITY
6. E46 – SOFT COMPUTING

Every Student is supposed to take one elective paper in Semester 4 as part of the MCA course. Examinations will be conducted for the selected paper by the University with both internal and external components like other core papers.
MCA 405 E41 - MICROPROCESSORS AND EMBEDDED SYSTEMS

UNIT I


UNIT II

Instruction Set, Assembler Directives and Assembly Language Programming of 8086 - Machine Language Instruction Formats – Instruction Set of 8086-Data transfer instructions, Arithmetic and Logic instructions, Branch instructions, Loop instructions, Processor Control instructions, Flag Manipulation instructions, Shift and Rotate instructions ,String instructions, Assembler Directives and operators, Example Programs, Introduction to Stack, STACK Structure of 8086, Interrupts and Interrupt Service Routines, Interrupt Cycle of 8086, Non-Maskable and Maskable Interrupts, Interrupt Programming, MACROS.

UNIT III

Special Purpose Programmable Devices and their Interfacing - Data transfer schemes-programmed I/O, Interrupt I/O, DMA, DMA Controller 8257, Programmable Interval Timer 8253, Programmable Interrupt Controller 8259A, Programmable Communication Interface 8251 USART, Programmable Peripheral Interface 8255.

UNIT IV

Architecture and Comparison of various Processors - 80186, 80286, 80386, 80486, Pentium Processors, Case Study on Advanced Multiprocessors

UNIT V

Introduction to Embedded Systems .


References

- The Intel Microprocessors 8086/8088, 80816/80188, 80286, 80486 Pentium and Pentium Pro Processor – Architecture, Programming and interfacing by Brey, 4th Edition, PHI.
- Microprocessor x86 Programming by K.R. Venugopal and Raj Kumar – BPB publications
MCA 405 E42 - BIG DATA ANALYTICS

UNIT I

UNIT II

UNIT III

UNIT IV

UNIT V

REFERENCES
MCA 405 E43 - CLOUD AND GRID COMPUTING

UNIT I


UNIT II


UNIT III


UNIT IV


UNIT V


REFERENCES

MCA 405 E44 - SOCIAL NETWORK ANALYSIS

UNIT I INTRODUCTION

UNIT II MODELLING, AGGREGATING AND KNOWLEDGE REPRESENTATION

UNIT III EXTRACTION AND MINING COMMUNITIES IN WEB SOCIAL NETWORKS

UNIT IV - PREDICTING HUMAN BEHAVIOUR AND PRIVACY ISSUES

UNIT V - VISUALIZATION AND APPLICATIONS OF SOCIAL NETWORKS

REFERENCE BOOKS
UNIT I

UNIT II

UNIT III

UNIT IV

UNIT V

REFERENCES
- Menezes, Network Security and Cryptography, Cengage Learning
MCA 405 E46 - SOFT COMPUTING

UNIT I

UNIT II
Artificial Neuron, Activation Functions, Training an Artificial Neural Net work- Supervised, Unsupervised, Reinforced Training, Single and multi layer networks, Basic learning laws, Perceptron convergence, Back Propogation Network, Associative memory, Competitive Learning, Self organizing neural networks- Kohonen self organizing Maps

UNIT III
Difference between Traditional Algorithms and GA, The basic concepts, creation of offspring, working principles, Encoding , Fitness function, Reproduction , Cross Over, Mutation, Convergence Theory, Applications.

UNIT IV
Evolutionary Computing, Simulated Annealing, Particle Swarm optimization, Real to Artificial Ants, Ant colony optimization, Ant Colony Optimization Algorithms for the Traveling Salesman Problem

UNIT V
Hybrid Systems: Neural-Network-Based Fuzzy Systems, Fuzzy Logic-Based Neural Networks, Genetic Algorithm for Neural Network Design and Learning, Fuzzy Logic and Genetic Algorithm for Optimization, Applications.

REFERENCES

- Neuro-Fuzzy and Soft Computing, Jang, Sun, & Mizutani, PHI.
- S. Rajasekaran & GA Vijayalakshmi Pai “Neural Networks, Fuzzy Logic, and Genetic Algorithms synthesis and application”, PHI
- James F. Kennedy, Russell C. Eberhart, Swarm intelligence, Morgan Kaufman, 2001
MCA 406P LINUX OS & SHELL PROGRAMMING PRACTICALS

Develop Programs to demonstrate the following concepts

Installation of Linux, network based installation,

Basic Overview of various commands- cal, pwd, cd, ls, mv, cd, cp, rm, mkdir, rmdir, more, less, touch.

Creating and viewing files using cat, file comparisons, disk related commands, checking disk free spaces.

Batch commands, kill, ps, who, Printing commands, find, sort, touch, file, file processing commands- wc, cut, paste etc. mathematical commands - expr, factor etc.

Filter commands- pr, head, tail, cut, sort, uniq, tr - Filter using regular expression grep, egrep, sed, awk.

Shell Programming -Shells, Scripting Rationale Creating a bash Script, bash Start up Files, A Script’s Environment, Exporting Variables, Exit Status, Programming the Shell, Parameter Passing, Operators, looping, Input and Output.

Process Management with Linux, File System management, User Administration, Linux Start up and Shutdown, Software package Management

Network Administration LAN Card configuration, Server Configuration- DHCP, DNS, FTP, Telnet, SSH, NFS, WebServer, SQUID Proxy server.

References

- Operating System - Linux, NUT Press, PHI Publisher, 2006 Edition
- Red Hat Linux Bible, Cristopher Negus, Wiley Dreamtech India
- UNIX Shell Programming by YeswantKanetkar, BPB
- Beginning Linux Programming by Neil Mathew & Richard Stones, Wiley Dreamtech India
The mini project is designed to help students develop practical ability and knowledge about practical tools/techniques in order to solve real life problems related to the industry, academic institutions and computer science research.

The course Mini Project is one that involves practical work for understanding and solving problems in the field of computing. Any computer science project usually consists of the following: analysis, design, coding/implementation and testing of some information system or subsystem, such as, a piece of software. In this course we expect a software system or subsystem.

This course will also develop your investigative, research and report writing skills and will provide an opportunity for you, to investigate a chosen topic in considerable depth. Mini Project provides the opportunity for students to demonstrate the application of their programming and research skills, and to apply their knowledge to complex computing problems.

**Project Team**

The project team should be organized and determined towards the fulfilment of their projects’ objectives and tasks. A maximum of two students should work on a project, however, an individual student can also undertake the project on his/her own.

The main responsibilities of the project team/student are to:

- Ensure that an appropriate amount of time and effort is applied to the project,
- Ensure that they are responsive to the guidance of their counsellor,
- Acknowledge the text, material and ideas of others properly,
- Meet all milestones and regulations related to the work, and
- To communicate any problems that are likely to prejudice the quality or time lines of the work to the counsellor as and when such problems arise.

**Project Categories**

Four broad areas / categories of computer science are given below, so that you can select any of these category for your Mini project.

- Application development
- Networking project
- System software
- Website development.
MCA 501T  USER INTERFACE DESIGN

UNIT I  Introduction
Introduction-Importance-Human-Computer interface-characteristics of graphical and web user interface-advantages and disadvantages of graphical systems-characteristics of GUI - web user interface-popularity-characteristics of Web interface-internet-intranet-extranet-principles of UID

UNIT II  Human Computer Interaction
User interface design process- obstacles-usability-Creating graphical systems-Know your user client-human characteristics in design-Understand the business functions-requirement analysis-Direct-Indirect methods-basic business functions-Design standards-system trainings—Understand the principles of good screen design-Human consideration in screen design-Develop System Menus and Navigation Schemes-structures of menus - functions of menus-contents of menu-formatting - phrasing the menu - selecting menu choice-navigating menus-graphical menus.

UNIT III  Windows

UNIT IV  Multimedia
Write clear text and messages-Text for web pages - effective feedback-guidance & assistance-Internationalization accessibility-Icons-Image-Multimedia-coloring.

UNIT V  Windows Layout - Test

References
MCA 502T KNOWLEDGE MANAGEMENT AND BUSINESS INTELLIGENCE

UNIT - I

UNIT - II

UNIT - III

UNIT - IV

UNIT - V

REFERENCES:
2. Efraim Turban, Ramesh Sharda, Dursun Delen and David King, “Business Intelligence” 2 nd Edition, 2010. (For Unit IV – Chapter 1, Unit – V -Chapter 6)03
MCA 503T ENTERPRISE RESOURCE PLANNING

MODULE I
INTRODUCTION - Overview of enterprise systems – Evolution - Risks and benefits - Fundamental technology - Issues to be consider in planning design and implementation of cross functional integrated ERP systems.

MODULE II
ERP SOLUTIONS AND FUNCTIONAL MODULES - Overview of ERP software solutions- Small, medium and large enterprise vendor solutions, BPR, and best business practices - Business process Management, Functional modules.

MODULE III

MODULE IV
POST IMPLEMENTATION - Maintenance of ERP- Organizational and Industrial impact; Success and Failure factors of ERP Implementation.

MODULE V
EMERGING TRENDS ON ERP - Extended ERP systems and ERP add-ons -CRM, SCM, Business analytics - Future trends in ERP systems-web enabled, Wireless technologies, cloud computing.

Case studies on ERP Systems used in Industries/Organizations

REFERENCES
4. Mahadeo Jaiswal and Ganesh Vanapalli, ERP Macmillan India, 2009
UNIT I - JAVA FUNDAMENTALS

UNIT II NETWORK PROGRAMMING IN JAVA

UNIT III APPLICATIONS IN DISTRIBUTED ENVIRONMENT

UNIT IV MULTI-TIER APPLICATION DEVELOPMENT

UNIT V ENTERPRISE APPLICATIONS

REFERENCES:
LIST OF ELECTIVES FOR SEMESTER 5

1. E51 – ADHOC AND SENSOR NETWORKS
2. E52 – MULTIMEDIA SYSTEMS
3. E53 – INFORMATION SECURITY AND E-COMMERCE
4. E54 – DIGITAL IMAGE PROCESSING
5. E55 – DISTRIBUTED COMPUTING
6. E56 – COMPUTER GRAPHICS WITH OPENGL

Every Student is supposed to take one elective paper in Semester 5 as part of the MCA course. Examinations will be conducted for the selected paper by the University with both internal and external components like other core papers.
UNIT - I


UNIT - II

TCP over mobile ad hoc networks: IP address acquisition, effects of partitions on TCP, provisions for mobility and fairness. Issues in designing a MAC Protocol- Classification of MAC Protocols- Contention based protocols- Contention based protocols with Reservation Mechanisms- Contention based protocols with Scheduling Mechanisms – Multi channel MAC-IEEE 802.11, Network simulators

UNIT - III


UNIT - IV

Hybrid wireless networks and wireless sensor networks: Architectures and routing protocols for hybrid wireless networks; Load balancing schemes; Pricing schemes for multihop wireless Networks, Single node architecture: hardware and software components of a sensor node – WSN Network architecture: typical network architectures-data relaying and aggregation strategies-MAC layer protocols: self-organizing, Hybrid TDMA/FDMA and CSMA based MAC-IEEE 802.15.4

UNIT - V

Recent advances in wireless networks Wide Band (UWB) communication; Issues and challenges in UWB communication; Applications of UWB communication; Wireless Fidelity (Wi-Fi) systems; Issues in Wi-Fi Systems; Pricing/billing in Wi-Fi systems; Multimode 802.11; Optical wireless communications; Optical Wireless Wavelength Division Multiplexing (OWWDM). Security in Ad Hoc and Sensor Networks, Integrating MANETs, WLANs and Cellular Networks

References

MCA 505  E52 - MULTIMEDIA SYSTEMS

UNIT I

Introduction, Characteristics, Hardware and Software Requirements, Applications of Multimedia. **Media and data Streams** - Medium, Properties of Multimedia System, Traditional Data Streams, Continuous Data Stream.

UNIT 2


UNIT 3


UNIT 4

**Multimedia DBMS**: Characteristics, Data Structure, Operations, Database Model, SGML, ODA, MHEG. **Synchronization**: – Notion of synchronization, presentation requirements, Synchronization case studies –MHEG, HyTime, MODE, ACME

UNIT 5

**Multimedia Application Development**: Design, Development and evaluation of multimedia a system - The development of user interface design. Multimedia & the Internet, Multimedia conferencing and file sharing, Multimedia broadcasting, Multimedia Development Issues, Multimedia project - Structured Multimedia development, Multimedia project timing

REFERENCES:

- Steinmetz & Nahrstedt, “Multimedia : Computing, Communications and Applications”, Pearson Education
- Mohammad Dastbaz, Designing Interactive Multimedia Systems
- Multimedia – Technology and applications David Hillman Galgotia Publications, Delhi
MCA 505 E53 -INFORMATION SECURITY AND E-COMMERCE

UNIT I

UNIT II

UNIT III

UNIT IV
Electronic Data Interchange: EDI applications in Business – EDI and e Commerce – EDI standardization and implementation – Internet based EDI

UNIT V

REFERENCES:
2. Ravi Kalakota, Andrew, “Frontiers of Electronic Commerce”, Addison Wesley
5. Information Systems Security, Godbole, Wiley-India
6. Information Security Principles and Practice, Deven Shah, Wiley-India
UNIT I

UNIT II

UNIT III

UNIT IV

UNIT V

REFERENCES:
- Prentice Hall of India, 2002.
MCA 505 E55 - DISTRIBUTED COMPUTING

UNIT I

UNIT II
Communication – RPC, Message oriented Communication-Transient communication, Persistent Communication-Message queuing model, Architecture of queuing system, Message Brokers, Stream oriented Communication

UNIT III
Synchronization – Clock synchronization, Logical Clocks, Mutual Exclusion, Election Algorithms-Bully Algorithm, Ring Algorithm Consistency and replication – Data centric consistency, client centric consistency, consistency protocols.

UNIT IV
Fault Tolerance – introduction, process resilience, reliable client-server communication, reliable group communication- Basic Reliable-Multicasting Schemes, Atomic Multicast, distributed commit.

UNIT V

REFERENCES:
- Elements of Distributed Computing, Garg, Wiley Publications
- Distributed Operating System, Pradeep K Sinha, PHI Publications
MCA 505 E56 - COMPUTER GRAPHICS WITH OPEN GL

UNIT I


UNIT II


UNIT III

Projection: 3D Geometric transformations- Translation, Scaling, Rotation. Perspective parallel Matrix representation – 3D viewing pipeline – 3D clipping

UNIT IV


UNIT V


REFERENCES

- FoleyJ.D., Andries Van Dam, Computer Graphics-Principles and Practice, Addison-Wesley.
- Computer Graphics using Open GLFS Hill-Prentice Hall
MCA 506P ADVANCED JAVA PROGRAMMING PRACTICALS

Students are directed to develop programs on demonstrating the theoretical concepts on Advanced Java. A minimum of 20 programs is required in this practical course.

Some tentative programs that can be experimented are as follows:

1. Development of dynamic website of an online Departmental Store. The website should be user friendly and should have the following pages: • Home page • Registration and user login • User profile page • Items catalog • Shopping cart • Payment by credit card • Order confirmation

2. Add validations to the above site for registration, user login, user profile and payment by credit card using Java Script.

3. Creation of a JavaBean which gives the converted value of Temperature (in degree celcius) into equivalent Fahrenheit

4. Creation of a simple Bean with a label – which is a “count” of number of clicks. Then create a BeanInfo class such that only the “count” is visible in the Property Window.

5. Creation of two Beans a) Keypad b) Display pad. After that integrate the two beans to make it work as a calculator.

6. Do the assignment 2 using JSP by converting the static web pages of assignment 2 into dynamic web pages. Create database with User Information and Item information. The Item catalog should be dynamically loaded from the database.

MCA 507P PYTHON Programming Practicals

UNIT I

UNIT II
Text Files: Strings, Formatted Output, Files, Errors and Exception Handling, Execution and Control Structures: if Statement, for Loop, Two Dimensional Lists, while Loop, More Loop Patterns, Additional Iteration Control Statements, Containers and Randomness:Dictionaries , Other Built-in Container Types, Character Encoding and Strings, Module random, Set Data Type.

UNIT III

UNIT IV

UNIT V

Reference
4. Dr.Varghese Paul, Dr.Anjana S.Chandran,“Introduction To Computing And Problem Solving Using Python”,Educational Publishers And Distributors, 2016
MAIN SEMINAR MCA508S – Semester V

Regulation for conducting Main Seminar [MCA508S]

The Main Seminar [MCA508S] of Semester 5 is intended to make MCA students aware of the Current / Future trends related to Information Technology/ Computer Science/ Computer Application. As such, a seminar report of not less than 15 pages is to be prepared and submitted for final evaluation.

The Main Seminar is evaluated internally by the College and carries a total Marks of 50 divided as follows:

1. Marks for relevance of topic and literature study (20)
2. Marks for Presentation – 2 presentations (20)

The seminar report should be prepared as per the following guidelines:
1. No of pages : Not less than 15
2. Size A4, One sided
3. Text Size 12 ; Title Size 14 Underlined; Line spacing : 1.5 Full Justified
4. Spiral Binding with uniformity in bind cover.

Every student is expected to present a minimum of 2 presentation of the seminar before the evaluation committee and for each presentation marks can be equally apportioned. A three member committee consisting of qualified TEACHERS With PG in Computer Science / Computer Application from the MCA Department has to be appointed by HoD. The Committee duly appointed will evaluate the seminar.

At the end of the semester the total marks have to be calculated and send to the University.

A Student shall have to score 50 % for getting a pass in the Seminar [MCA508S].
The Master of Computer Applications (MCA) programme prepares the students to take up positions as Systems Analysts, Systems Designers, Software Engineers, Programmers and Project Managers in any field related to information technology. As part of the curriculum, all students who are into their sixth semester will have to carryout a project preferably in a software industry or any research organization for duration of one full semester. The courses studied and the mini project & the main project handled at final year will give the comprehensive background to work on diverse application domains.

The objective of the MCA project work is to develop quality software solution. During the development of the project, the student should involve in all the stages of the software development life cycle like requirements engineering, systems analysis, systems design, software development, testing strategies and documentation with an overall emphasis on the development of reliable software systems. The primary emphasis of the project work is to understand and gain the knowledge of the principles of software engineering practices, so as to participate and manage a large software engineering projects in future. Students should take this project work very seriously, and carry out the same individually. The topics selected should be complex and large enough to justify as an MCA project. The project should be genuine and original in nature and should not be copied from anywhere else.

After the completion of this project work, the student should be able to:

- Describe the Systems Development Life Cycle (SDLC).
- Evaluate systems requirements.
- Complete a problem definition and its evaluations.
- Construct and evaluate UML's/Data flow diagrams and Data Dictionaries
- Evaluate alternative tools for the analysis process.
- Create and evaluate such alternative graphical tools as systems flow charts and state transition diagrams.
- Plan the systems design phase of the SDLC.
- Distinguish between logical and physical design requirements.
- Design and evaluate system Inputs & outputs and UI.
- Decide various data structures.
- Perform coding for the project.
- Documentation requirements and prepare and evaluate systems documentation.
- Generate various reports.
- To decide the future scope and further enhancement of the system.
- Develop of the ability to assess the implications of work performed.
- Get good exposure and command in one or more application areas and on the software
- Develop of the ability to communicate effectively.

All students are expected to work on a real-life project preferably in some Industry / Research and Development Laboratories / IT-ITES Organisations. The complete project work should be done by the student only. The role of guide should be about guidance wherever any problem encounters during project.

- Not more than one student is permitted to work on a project.
- Each Student should be involved in each and every phase of Project Development. If it is found that student is not involved in any phase; for example coding phase, it may lead to the rejection/disqualifying of the project at any stage.
- Title of the project should be kept the same throughout the project.
Guidelines for preparing the Project Dissertation

This document lists the contents required for the academic project report done as part of the MCA Curriculum. Section names have been listed with description. The descriptions have been provided in italics. Important: This page and the text in italics present throughout this document are to give you guidance. Please do not include them in your project report.

CONTENTS OF THE ACADEMIC PROJECT REPORT

1. Cover Page as per format
   Use the same format given in the project doc

2. Certificate of the Company/Organization
   Use the same format given in the project doc

3. Certificate of the Head Of Department as per format
   Use the same format given in the project doc

4. Certificate of the Internal project guide as per format
   Use the same format given in the project doc

5. Declaration
   By student – format given

6. Acknowledgement
   Use the same format given in the project doc

7. Revision history
   Table with version, date, author, changes done, approval

8. Table of Contents
   Please use the MS Word Table of content feature for this and not a manual TOC.

9. Executive Summary
   This should describe the problem and the solution given by your project in brief. You should also mention the process model you used for development, methodology and technology. Limit the description to 1-2 pages.

10. Background
    UP Phase: Inception
    10.1. Existing System
    Describe the system that already exists. Please note that the system could be manual or automated or a combination of both. Provide the business flow using an activity diagram.
    10.2. Definition of Problem-
    Describe the problems/inadequacies of current set up.
    10.3 Proposed System
    Explain how the proposed system will solve the problems.
    Provide the revised business flow involving your system using an activity diagram, if relevant.

11. Project Overview
    UP Phase: Inception
    11.1. Objective of the Project
    Describe the business benefits expected from this project.
    11.2. Stakeholders
    List the stakeholders, their goals which will be satisfied by this system and the benefits.
    11.3. Scope of the Project
    Mention in brief the system proposed to meet the objective. Mention clearly if any part of the work is not in your scope – e.g. installation, or some data migration required for implementation of this system, integration with some other system etc.
    11.4 Feasibility Analysis
    11.4.1. Technical feasibility
    Technology and system feasibility - The assessment is based on an outline design of system requirements in terms of Input, Processes, Output, Fields, Programs, and Procedures. This can be quantified in terms of volumes of data, trends, frequency of updating, etc. in order to estimate whether the new system will perform adequately or not
    11.4.2. Operational feasibility
Is a measure of how well a proposed system solves the problems, and takes advantages of the opportunities identified during scope definition and how it satisfies the requirements identified in the requirements analysis phase of system development

11.4.3. Schedule feasibility
Schedule feasibility is a measure of how reasonable the project timetable is.

11.5.4. Economic feasibility - Cost - Benefit Analysis
If the company had already done the analysis before deciding on the project, then obtain the information from the company and just mention it here. If this analysis was performed by the student, then explanation of how the analysis was conducted should be provided. The analysis aims to determine the benefits and savings that are expected from a candidate system and compare them with costs.

12. Overall Project Planning

12.1. Development environment
Identify and list the technology and tools planned to be used in the development of the project – IDEs, compilers, UML tool, configuration management tool etc.

12.2. Constraints
List the constraints applicable to your project e.g. time constraints, budget constraints, resource constraints, any other constraints set by the customer.

12.3. Deliverables
List all deliverables expected by the customer – e.g. application, configuration files, source code, any 3rd party software packaged along with this, documents – requirements, design, user manual, installation manual, tutorial – as applicable for your project.

12.4. Assumptions and dependencies
List the assumptions and dependencies made while planning the project, eg. there could be an assumption that the required inputs will be given by users or that users will be available to review the documents or that the testing team will be available to test if required etc.

12.5. Risks
List any risks you foresee in the execution of this project and describe the plan to mitigate it.

12.6. Process model
Mention which process model you chose to develop this project and the justification for it.

12.7. Test Strategy
Mention all testing strategies – unit, integration, system, user acceptance testing that have been used.

12.8. Testing environment and tools
The environment for testing should give the specifications of hardware and software used for testing. Tools used for testing if any, should be listed here.

13. Iteration Planning

13.1. Schedule
Put the schedule here – activities, dependencies, start and end dates.

13.2. Risks
List any risks you foresee in the execution of this iteration and describe the plan to mitigate it.

14. High level system Analysis

14.1. User characteristics
Mention the different types of users or user groups of the proposed system and any special training needs they have in order to use this system.

14.2. Summary of system features/Functional requirements
List the features identified to be part of this system in order to satisfy this goal.

14.3. Non Functional Requirements / Supplementary Specification
List the non functional requirements applicable to your project related to performance, security etc.

14.4. Glossary
Define the business/domain terms specific to the context of this system
14.5. **Business Rules**

Define the business rules specific to the user’s domain/organization that need to be satisfied by this system.

14.6. **Use cases**

List all the use case names here and a brief description of each use case.

14.7. **Use case diagram**

15. **Domain Model**

UP Phases: Inception and Elaboration

The initial model will be identified during elaboration and will be continued to be revised in the elaboration stage. Give the analysis level class diagram, i.e. the domain model here and a brief description of the analysis level classes.

16. **Use Case Model**

Relevant UP Phases: Inception and Elaboration

To be developed during inception for a critical few use cases and the remaining use cases (majority) will be developed during elaboration for the majority of the use cases.

16.1. **Use case text**

Write the detailed use case text, in the fully dressed format for each use case. Identify the non functional requirements and rules to be followed specific to the use case being considered. Also identify the user inputs in the form of text and file inputs to system (if any). Develop the format of input which the system will accept, with examples. If it is a file input, format of file and format of data within it (if relevant) have to be provided.

Develop error messages and information texts required as part of this use case. Any external system interface requirements also need to be identified.

16.2. **System sequence diagram**

For each use case under consideration, draw the system sequence diagram(s).

16.3. **Operation contracts**

For each system sequence diagram under consideration, write up the operation contracts.

16.4. **Reports**

Develop the format of the reports generated as part of this use case, if any.

16.5. **Sequence diagrams**

Develop the design level sequence diagrams for the use case under consideration.

16.6. **Class diagrams**

Develop the design level class diagram for the use case under consideration.

16.7. **UI design**

Develop the screens identified for the use case under consideration and provide snapshots. At this stage, static screens are sufficient.

16.8. **Theoretical Background**

Theoretical details about the technology, tools and algorithms you have used in this project should be mentioned here in brief.

16.9. **Architecture**

In this section, show pictorially the logical and deployment architecture of this system. Use package diagrams, component and deployment diagrams for this.

16.10. **Database design**

This should give a catalogue of the data elements used in the system / sub system developed. The following are the details require for each table and field in the table. Repeat this list as many times there are tables and fields. Write NA if NOT applicable:

16.10.1. **Table Name**

16.10.1.1. **Field Name**

16.10.1.2. **Length**

11.4.1.1. **Type** CHAR, VARCHAR, NUMBER, DATE etc.

11.4.1.2. **Description**
17. Testing
UP Phases: Construction and Transition
Developed completely in construction and revised in transition phase.

17.1. Test cases
List each test case – with description, inputs, expected output, pass/fail criteria.

17.2. Test Report
Actual result against the expected results of test cases should be compiled here. A measure of quality like % of passed test cases should also be provided.

17.3. Sample Code used for testing
Sample code used for unit testing should be provided.

18. Transition
Relevant UP Phase: Transition

18.1. System Implementation
Describe the implementation mechanisms. Describe the method of data conversion and migration for the new system if applicable.

18.2. System Maintenance
Describe the plan for maintenance of the system. Mention the documents and any training provided by the student for future maintenance.

18.3. User/Operational Manual
If there was a user manual expected as deliverable by the customer, provide it here. If there was a demo or training given to users on the system, mention that. If there are any limitations of the system or constraints on inputs like data format, which have to be taken care by users, list it here. Also mention the details required for operation of the system. This should include instructions on how to start and shutdown the system, description of expected folder structure of system related files after installation, list of roles of users required to be created and maintained in the system.

If there are any requirements to do periodic cleaning of data, those have to be mentioned here. If the delivery of scripts or programs for automatic data cleaning is in scope, usage of the scripts should be described. Configuration management related information, if applicable, should be provided to suggest frequency of backups of files.

19. Annexure:

19.1. Organization profile
Give a brief background of the organization where the student has developed the project

19.2. Document Glossary, Figures, Tables
List of abbreviations should be provided in the document glossary. Each figure and table should be labeled. You should create an index for these like the table of contents.

19.3. References:
Books: Any references you made to books and papers should be listed here with the book name, edition, name of author and publisher.
Websites: Any references you made to websites should be listed here with the URL and date of access.

19.4. User Interview Questionnaires

19.5. Sample Project code / Algorithm if project code is not available.

The format of various certificates to be included in the Project report is appended along with this guidelines.
Format of certificates to be attached in the project report

A Project Report

On

“PROJECT TITLE”

Submitted to the

Department of MCA

In partial fulfillment of the

MASTER OF COMPUTER APPLICATIONS

Under the guidance of

Internal Guide’s Name

Project Done by

NAME OF STUDENT
(Reg No: )

EMBLEM OF COLLEGE

DEPARTMENT OF MCA

NAME AND ADDRESS OF COLLEGE

Month-Year
BONAFIDE CERTIFICATE

Certified that the Project Work entitled

“PROJECT TITLE”

is a bonafide work done by

Name of the student

In partial fulfillment of the requirement for the Award of

MASTER OF COMPUTER APPLICATIONS

Degree From

Mahatma Gandhi University, Kottayam

(Period of study)

Head of Department
Project Guide

Submitted for the Viva-Voce Examination held on……………………………

External Examiner1
(Name & Signature)

External Examiner2
(Name & Signature)
This is to certify that the project entitled “PROJECT TITLE” has been successfully carried out by NAME OF STUDENT (Reg. No:) in partial fulfilment of the Course Master of Computer Applications.

INTERNAL GUIDE

Date: HEAD OF THE DEPARTMENT
This is to certify that the project entitled “PROJECT TITLE” has been successfully carried out by NAME OF STUDENT (Reg no:) in partial fulfilment of the course Master of Computer Applications under my guidance.

Date:                                      Name of Guide

INTERNAL GUIDE
DECLARATION

1. NAME OF STUDENT, hereby declare that the project work entitled “NAME OF THE PROJECT” is an authenticated work carried out by me at XYZ SOFTWARE PVT. LTD. under the guidance of Guide’s Name for the partial fulfilment of the course MASTER OF COMPUTER APPLICATIONS. This work has not been submitted for similar purpose anywhere else except to NAME OF COLLEGE.

I understand that detection of any such copying is liable to be punished in any way the school deems fit.

NAME OF STUDENT

Signature

Date:

Place:
The Viva-Voce Examination of 6th Semester is a comprehensive evaluation of what has been learned through the entire MCA programme.

Students will be evaluated through all core subjects of the MCA programme and marks will be awarded on the basis of oral answers given by the student.

There is no internal mark component for the same. The maximum marks for the Viva Voce examination is 100. The evaluation is done by the evaluators duly appointed by the University.