## MCA 1st Year (I-Semester)

<table>
<thead>
<tr>
<th>Code</th>
<th>Subject</th>
<th>L</th>
<th>P</th>
<th>C</th>
<th>Int.</th>
<th>Ext.</th>
<th>Total</th>
<th>Ext.</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>MC 1.1</td>
<td>Discrete Structures</td>
<td>4</td>
<td>-</td>
<td>8</td>
<td>40</td>
<td>60</td>
<td>100</td>
<td>24</td>
<td>50</td>
</tr>
<tr>
<td>MC 1.2</td>
<td>Computer Organization</td>
<td>4</td>
<td>-</td>
<td>8</td>
<td>40</td>
<td>60</td>
<td>100</td>
<td>24</td>
<td>50</td>
</tr>
<tr>
<td>MC 1.3</td>
<td>Object Oriented Programming (through C++)</td>
<td>4</td>
<td>-</td>
<td>8</td>
<td>40</td>
<td>60</td>
<td>100</td>
<td>24</td>
<td>50</td>
</tr>
<tr>
<td>MC 1.4</td>
<td>Probability and Statistics</td>
<td>4</td>
<td>-</td>
<td>8</td>
<td>40</td>
<td>60</td>
<td>100</td>
<td>24</td>
<td>50</td>
</tr>
<tr>
<td>MC 1.5</td>
<td>Accountancy and Financial Management</td>
<td>4</td>
<td>-</td>
<td>8</td>
<td>40</td>
<td>60</td>
<td>100</td>
<td>24</td>
<td>50</td>
</tr>
<tr>
<td>MC 1.6</td>
<td>English Language Communication Skills</td>
<td>2</td>
<td>-</td>
<td>4</td>
<td>20</td>
<td>30</td>
<td>50</td>
<td>12</td>
<td>25</td>
</tr>
<tr>
<td>Practical</td>
<td>English Language Communication Skills Lab</td>
<td>-</td>
<td>2</td>
<td>2</td>
<td>50</td>
<td></td>
<td>50</td>
<td>25</td>
<td>25</td>
</tr>
<tr>
<td>MC 1.7</td>
<td>Computer Organization Lab</td>
<td>-</td>
<td>4</td>
<td>4</td>
<td>40</td>
<td>60</td>
<td>100</td>
<td>24</td>
<td>50</td>
</tr>
<tr>
<td>MC 1.8</td>
<td>Object Oriented Programming (through C++) Lab</td>
<td>-</td>
<td>4</td>
<td>4</td>
<td>40</td>
<td>60</td>
<td>100</td>
<td>24</td>
<td>50</td>
</tr>
</tbody>
</table>

## MCA 1st Year (II-Semester)

<table>
<thead>
<tr>
<th>Code</th>
<th>Subject</th>
<th>L</th>
<th>P</th>
<th>C</th>
<th>Int.</th>
<th>Ext.</th>
<th>Total</th>
<th>Ext.</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>MC 2.1</td>
<td>Data Structures (through Java)</td>
<td>4</td>
<td>-</td>
<td>8</td>
<td>40</td>
<td>60</td>
<td>100</td>
<td>24</td>
<td>50</td>
</tr>
<tr>
<td>MC 2.2</td>
<td>Operating Systems</td>
<td>4</td>
<td>-</td>
<td>8</td>
<td>40</td>
<td>60</td>
<td>100</td>
<td>24</td>
<td>50</td>
</tr>
<tr>
<td>MC 2.3</td>
<td>Organizational Structure and Personal Management</td>
<td>4</td>
<td>-</td>
<td>8</td>
<td>40</td>
<td>60</td>
<td>100</td>
<td>24</td>
<td>50</td>
</tr>
<tr>
<td>MC 2.4</td>
<td>Operations Research</td>
<td>4</td>
<td>-</td>
<td>8</td>
<td>40</td>
<td>60</td>
<td>100</td>
<td>24</td>
<td>50</td>
</tr>
<tr>
<td>MC 2.5</td>
<td>Business Data Processing</td>
<td>4</td>
<td>-</td>
<td>8</td>
<td>40</td>
<td>60</td>
<td>100</td>
<td>24</td>
<td>50</td>
</tr>
<tr>
<td>Practical</td>
<td>Data Structures (through Java) Lab</td>
<td>-</td>
<td>4</td>
<td>4</td>
<td>40</td>
<td>60</td>
<td>100</td>
<td>24</td>
<td>50</td>
</tr>
<tr>
<td>MC 2.6</td>
<td>Business Data Processing Lab</td>
<td>-</td>
<td>4</td>
<td>4</td>
<td>40</td>
<td>60</td>
<td>100</td>
<td>24</td>
<td>50</td>
</tr>
</tbody>
</table>
## MCA 2nd Year (III-Semester)

<table>
<thead>
<tr>
<th>Code</th>
<th>Subject</th>
<th>L</th>
<th>P</th>
<th>C</th>
<th>Int.</th>
<th>Ext.</th>
<th>Total</th>
<th>Ext.</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>MC 3.1</td>
<td>Database Management Systems</td>
<td>4</td>
<td>-</td>
<td>8</td>
<td>40</td>
<td>60</td>
<td>100</td>
<td>24</td>
<td>50</td>
</tr>
<tr>
<td>MC 3.2</td>
<td>Computer Communications</td>
<td>4</td>
<td>-</td>
<td>8</td>
<td>40</td>
<td>60</td>
<td>100</td>
<td>24</td>
<td>50</td>
</tr>
<tr>
<td>MC 3.3</td>
<td>Unix Network Programming</td>
<td>4</td>
<td>-</td>
<td>8</td>
<td>40</td>
<td>60</td>
<td>100</td>
<td>24</td>
<td>50</td>
</tr>
<tr>
<td>MC 3.4</td>
<td>Management Information Systems</td>
<td>4</td>
<td>-</td>
<td>8</td>
<td>40</td>
<td>60</td>
<td>100</td>
<td>24</td>
<td>50</td>
</tr>
<tr>
<td>MC 3.5</td>
<td>Computer Graphics</td>
<td>4</td>
<td>-</td>
<td>8</td>
<td>40</td>
<td>60</td>
<td>100</td>
<td>24</td>
<td>50</td>
</tr>
<tr>
<td>MC 3.6</td>
<td>Database Management Systems Lab</td>
<td>-</td>
<td>4</td>
<td>4</td>
<td>40</td>
<td>60</td>
<td>100</td>
<td>24</td>
<td>50</td>
</tr>
<tr>
<td>MC 3.7</td>
<td>Unix Network Programming Lab</td>
<td>-</td>
<td>4</td>
<td>4</td>
<td>40</td>
<td>60</td>
<td>100</td>
<td>24</td>
<td>50</td>
</tr>
</tbody>
</table>

## MCA 2nd Year (IV-Semester)

<table>
<thead>
<tr>
<th>Code</th>
<th>Subject</th>
<th>L</th>
<th>P</th>
<th>C</th>
<th>Int.</th>
<th>Ext.</th>
<th>Total</th>
<th>Ext.</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>MC 4.1</td>
<td>Software Engineering</td>
<td>4</td>
<td>-</td>
<td>8</td>
<td>40</td>
<td>60</td>
<td>100</td>
<td>24</td>
<td>50</td>
</tr>
<tr>
<td>MC 4.2</td>
<td>Advanced Java Programming</td>
<td>4</td>
<td>-</td>
<td>8</td>
<td>40</td>
<td>60</td>
<td>100</td>
<td>24</td>
<td>50</td>
</tr>
<tr>
<td>MC 4.3</td>
<td>Data Warehousing and Mining</td>
<td>4</td>
<td>-</td>
<td>8</td>
<td>40</td>
<td>60</td>
<td>100</td>
<td>24</td>
<td>50</td>
</tr>
<tr>
<td>MC 4.4</td>
<td>Elective – I</td>
<td>4</td>
<td>-</td>
<td>8</td>
<td>40</td>
<td>60</td>
<td>100</td>
<td>24</td>
<td>50</td>
</tr>
<tr>
<td>MC 4.5</td>
<td>Elective – II</td>
<td>4</td>
<td>-</td>
<td>8</td>
<td>40</td>
<td>60</td>
<td>100</td>
<td>24</td>
<td>50</td>
</tr>
<tr>
<td>MC 4.6</td>
<td>Advanced Java Programming Lab</td>
<td>-</td>
<td>4</td>
<td>4</td>
<td>40</td>
<td>60</td>
<td>100</td>
<td>24</td>
<td>50</td>
</tr>
<tr>
<td>MC 4.7</td>
<td>Data Warehousing and Mining Lab</td>
<td>-</td>
<td>4</td>
<td>4</td>
<td>40</td>
<td>60</td>
<td>100</td>
<td>24</td>
<td>50</td>
</tr>
</tbody>
</table>

**MC4.4 Elective – I**
- MC4.4.1 Embedded Systems
- MC4.4.2 Human Computer Interaction
- MC 4.4.3 Perl Programming

**MC4.5 Elective – II**
- MC4.5.1 Distributed Operating Systems
- MC4.5.2 Mobile Computing
- MC4.5.3 Systems Programming
### MCA 3rd Year (V-Semester)

#### Scheme of evaluation

<table>
<thead>
<tr>
<th>Code</th>
<th>Subject</th>
<th>L</th>
<th>P</th>
<th>C</th>
<th>Int.</th>
<th>Ext.</th>
<th>Total</th>
<th>Ext.</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>MC 5.1</td>
<td>E-Commerce</td>
<td>4</td>
<td>-</td>
<td>8</td>
<td>40</td>
<td>60</td>
<td>100</td>
<td>24</td>
<td>50</td>
</tr>
<tr>
<td>MC 5.2</td>
<td>Multimedia Application Development</td>
<td>4</td>
<td>-</td>
<td>8</td>
<td>40</td>
<td>60</td>
<td>100</td>
<td>24</td>
<td>50</td>
</tr>
<tr>
<td>MC 5.3</td>
<td>Object Oriented Analysis and Design (using UML)</td>
<td>4</td>
<td>-</td>
<td>8</td>
<td>40</td>
<td>60</td>
<td>100</td>
<td>24</td>
<td>50</td>
</tr>
<tr>
<td>MC 5.4</td>
<td>Elective-III</td>
<td>4</td>
<td>-</td>
<td>8</td>
<td>40</td>
<td>60</td>
<td>100</td>
<td>24</td>
<td>50</td>
</tr>
<tr>
<td>MC 5.5</td>
<td>Elective – IV</td>
<td>4</td>
<td>-</td>
<td>8</td>
<td>40</td>
<td>60</td>
<td>100</td>
<td>24</td>
<td>50</td>
</tr>
<tr>
<td></td>
<td><strong>Practicals</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MC 5.6</td>
<td>UML Lab</td>
<td>-</td>
<td>4</td>
<td>4</td>
<td>40</td>
<td>60</td>
<td>100</td>
<td>24</td>
<td>50</td>
</tr>
<tr>
<td>MC 5.7</td>
<td>Multimedia Application Development Lab</td>
<td>-</td>
<td>4</td>
<td>4</td>
<td>40</td>
<td>60</td>
<td>100</td>
<td>24</td>
<td>50</td>
</tr>
</tbody>
</table>

**MC5.4 Elective – III**
- MC5.4.1 Software Project Management
- MC5.4.2 Middle Ware Technologies
- MC5.4.3 Information Retrieval Systems

**MC5.5 Elective – IV**
- MC5.5.1 Software Testing Methodologies
- MC5.5.2 Advanced Databases
- MC5.5.3 Information Security

### MCA 3rd Year (VI-Semester)

#### Scheme of evaluation

<table>
<thead>
<tr>
<th>Code</th>
<th>Subject</th>
<th>Int.</th>
<th>Ext.</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>MC6.1</td>
<td>Project Seminar</td>
<td>--</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>MC6.2</td>
<td>Dissertation/Thesis</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Excellent/Good/Satisfactory/Not-Satisfactory</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
I Year MCA             I  Semester

MC1.1 DISCRETE STRUCTURES

UNIT-I:

UNIT-II:

UNIT-III:

UNIT-IV:
Algebraic structures: Algebraic systems Examples and general properties, Semi groups and monads, groups sub groups’ homomorphism, Isomorphism.

UNIT-V:
Elementary Combinatorics: Basis of counting, Combinations & Permutations, with repetitions, Constrained repetitions, Binomial Coefficients, Binomial Multinomial theorems, the principles of Inclusion – Exclusion.

UNIT-VI:
Recurrence Relation: Generating Functions, Function of Sequences Calculating Coefficient of generating function, Recurrence relations, Solving recurrence relation by substitution and Generating funds. Characteristics roots solution of In homogeneous Recurrence Relation.

UNIT-VII:
Graph Theory: Representation of Graph, DFS, BFS, Spanning Trees, planar Graphs

UNIT-VIII:
Graph Theory and Applications, Basic Concepts Isomorphism and Sub graphs, Multi graphs and Euler circuits, Hamiltonian graphs, Chromatic Numbers
TEXT BOOKS:


2. Discrete Mathematical Structures with applications to computer science Trembly J.P. & Manohar .P, TMH


REFERENCE BOOKS:

1. Discrete Mathematical Structures, Bernard Kolman, Robert C. Busby, Sharn Cutter Ross, Pearson Education/PHI.

2. Discrete Mathematics for Computer science, Garry Haggard and others, Thomson

3. Discrete Mathematical Structures, Mallik and Sen, Thomson

4. Mathematical Foundations of computer science Dr D.S.Chandrasekharaiha Prism books Pvt Ltd.

UNIT I:

UNIT II:
COMBINATIONAL AND SEQUENTIAL CIRCUITS- decoders, Encoders, Multiplexers, Half and Full adders, Shift registers; Sequential circuits- flip-flops.

UNIT III:
MEMORY ORGANIZATION-memory hierarchy, Main memory-RAM, ROM chips, Memory address map, memory contention to CPU; Associative Memory-Hardware logic, match, read and write logic; Cache Memory-Associative mapping, Direct mapping, set-associative mapping, hit and miss ratio;

UNIT IV:
BASIC CPU ORGANIZATION-instruction formats-INTEL-8086 CPU architecture-Addressing modes - generation of physical address- code segment registers, Zero, one, two, and three address instructions.

UNIT V:
INTEL 8086 ASSEMBLY LANGUAGE INSTRUCTIONS-Data transfer instructions-input-output instructions, address transfer, Flag transfer, arithmetic, logical, shift, and rotate instructions.

UNIT VI:
INTEL 8086 ASSEMBLY LANGUAGE INSTRUCTIONS- conditional and unconditional transfer, iteration control, interrupts and process control instructions, assembler directives. Programming with assembly language instructions.
UNIT VII:
MICRO PROGRAMMED CONTROL: Control memory, Address sequencing, microprogram example, design of control unit Hard wired control. Microprogrammed control

UNIT VIII:
INPUT -OUTPUT ORGANIZATION-peripheral devices, input-output interface-I/O Bus and interface modules, I/O versus Memory bus, isolated versus memory mapped I/O, Modes of transfer-Programmed I/O, Interrupt-initiated I/O, priority interrupts-Daisy chaining, parallel priority, interrupt cycle, DMA- DMA control, DMA transfer, Input output processor-CPU-IOP communication.

TEXT BOOKS:


REFERENCE BOOKS:


    David A. Paterson and john L.Hennessy-Computer- organization and
design-Harcourt Asia Pte Ltd.-2nd Edition

    Structured Computer Organization – Andrew S. Tanenbaum, 4th Edition PHI/Pearson
UNIT-I
Different paradigms for problem solving, need for OOP, differences between OOP and Procedure oriented programming, Abstraction, Overview of OOP principles, Encapsulation, Inheritance and Polymorphism.

UNIT-II
C++ Basics: Structure of a C++ program, Data types, Declaration of variables, Expressions, Operators, Operator Precedence, Evaluation of expressions, Type conversions, Pointers, Arrays, Pointers and Arrays, Strings, Structures, References. Flow control statement- if, switch, while, for, do, break, continue, goto statements. Functions-Scope of variables, Parameter passing, Default arguments, inline functions, Recursive functions, Pointers to functions. Dynamic memory allocation and deallocation operators-new and delete, Preprocessor directives.

UNIT-III
C++ Classes And Data Abstraction: Class definition, Class structure, Class objects, Class scope, this pointer, Friends to a class, Static class members, Constant member functions, Constructors and Destructors, Dynamic creation and destruction of objects, Data abstraction, ADT and information hiding.

UNIT-IV
Polymorphism: Function overloading, Operator overloading, Generic programming-necessity of templates, Function templates and class templates.

UNIT-V
Inheritance: Defining a class hierarchy, Different forms of inheritance, Defining the Base and Derived classes, Access to the base class members, Base and Derived class construction, Destructors, Virtual base class.

UNIT-VI
Virtual Functions And Polymorphism: Static and Dynamic bindings, Base and Derived class virtual functions, Dynamic binding through virtual functions, Virtual function call mechanism, Pure virtual functions, Abstract classes, Implications of polymorphic use of classes, Virtual destructors.
UNIT-VII
C++ I/O: I/O using C functions, Stream classes hierarchy, Stream I/O, File streams and String streams, Overloading << and >> operators, Error handling during file operations, Formatted I/O.

UNIT-VIII
Exception Handling: Benefits of exception handling, Throwing an exception, The try block, Catching an exception, Exception objects, Exception specifications, Stack unwinding, Rethrowing an exception, Catching all exceptions, Design issues in exception handling.

TEXT BOOKS:

REFERENCE BOOKS:
MC1.4 PROBABILITY AND STATISTICS

UNIT-I:
**Probability:** Sample space and events – Probability – The axioms of probability - Some elementary theorems - Conditional probability – Baye’s theorem.

UNIT-II:

UNIT-III:
**Distribution** - Binomial, poisson and normal distribution – related properties.

UNIT-IV:
**Sampling distribution:** Populations and samples - Sampling distributions of mean (known and unknown) proportions, sums and differences.

UNIT-V:
**Estimation:** Point estimation – interval estimation - Bayesian estimation.

UNIT-VI:
**Test of Hypothesis** – Means and proportions – Hypothesis concerning one and two means – Type I and Type II errors. One tail, two-tail tests.

UNIT-VII:
**Tests of significance** – Student’s t-test, F-test, $\chi^2$ test. Estimation of proportions.

UNIT-VIII:
**Curve fitting:** The method of least squares – Inferences based on the least squares estimations - Curvilinear regression – multiple regressions – correlation for univariate and bivariate distributions.

TEXT BOOKS:
1. Probability and statistics for engineers:Erwin Miller And John E.Freund.
   Prentice-Hall of India / Pearson , Sixth edition.
REFERENCE BOOKS:


UNIT I:
Accounting: Principles, concepts, conventions, double entry system of accounting, introduction of basic books of accounts ledgers.

UNIT II:
Preparation of trial balance - Final accounts - company final accounts.

UNIT III:
Financial Management - meaning and scope, role, objectives of time value of money - over vitalization - under capitalization - profit maximization - wealth maximization - EPS maximization.

UNIT IV:
Ratio Analysis - advantages - limitations - Fund flow analysis - meaning, importance, preparation and interpretation of Funds flow and cash flow statements - statement of changes in working capital.

UNIT V:
Costing - nature and importance and basic principles. Absorption costing vs. marginal costing - Financial accounting vs. cost accounting vs. management accounting.

UNIT VI:
Marginal costing and Break-even Analysis: nature, scope and importance - practical applications of marginal costing, limitations and importance of cost - volume, profit analysis.

UNIT VII:
Standard costing and budgeting: nature, scope and computation and analysis - materials variance, labor variance and sales variance - budgeting - cash budget, sales budget - flexible Budgets, master budgets.

UNIT VIII:
Introduction to computerized accounting system: coding logic and codes, master files, transaction files, introduction documents used for data collection, processing of different files and Outputs obtained.
TEXT BOOKS:

2. Financial Accounting, S.N. Maheshwari, Sultan Chand Company
3. Financial Management, S.N. Maheshwari, Sultan Chand Company
Unit –I:
Features of Indian English - Correction of sentences - Structures - Tenses - ambiguity - idiomatic distortions.

UNIT-II:
Informal conversation Vs Formal expression Verbal and non-verbal communication, barriers to effective communication – kinesics

UNIT - III:
Types of Communication - Oral, aural, Writing and reading - Word-Power - Vocabulary- Jargon - rate of speech, pitch, tone - Clarity of voice

UNIT - IV:
Technical presentations - types of presentation – video conferencing-- participation in meetings - chairing sessions.

UNIT - V:
Formal and informal interviews – ambiance and polemics - interviewing in different settings and for different purposes e.g., eliciting and giving information, recruiting, performance appraisal.

UNIT - VI:
Written communication - differences between spoken and written communication - features of effective writing such "as clarity, brevity, appropriate tone clarity, balance etc.- GRE. TOEFL models


REFERENCES BOOKS:

Raymond V. Lesikav; John D. Pettit Jr.; Business Communication; Theory & Application, All India Traveller Bookseller, New Delhi-51.
Business Communication, RK Madhukar, Vikas Publishing House Pvt Ltd
Cliffs Test Prep for GRE and TOEFL: Computer Based Test, IDG Books. India (P) Ltd. New Delhi-002.
GRE and TOEFL; Kaplan and Baron's English in Mind, Herbert Puchta and Jeff Stranks, Cambridge
MC 1.7 English Language Communication Skills Lab

Objectives:
The language Lab focuses computer-aided multi-media instruction and language acquisition to achieve the following targets:

- To expose the students to a variety of self-instructional, learner-friendly modes of language learning.
- To help the students cultivate the habit of reading passages from the computer monitor, thus providing them with the required facility to face computer-based competitive exams such as GRE, TOEFL, GMAT etc.
- To enable them to learn better pronunciation through stress on word accent, intonation, and rhythm.
- To train them to use language effectively to face interviews, group discussions, public speaking.
- To initiate them into greater use of the computer in resume preparation, report writing, format-making etc.

However, depending upon the available infrastructure and budget, the above targets can also be achieved by procuring the minimum required equipment suggested for the establishment of a Conventional Lab the details of which are given below. The lab should cater to the needs of the students to build up their confidence to help them develop leadership qualities through their communicative competence.

ENGLISH LANGUAGE LABORATORY PRACTICE

Syllabus
The following course content is prescribed for the English Language Laboratory Practice:

- Introduction to Phonetics.
- Introduction to Vowels and Consonants and associated Phonetic symbols.
- Introduction to Accent, Intonation and Rhythm.
- Situational Dialogues / Role Play.
- Public Speaking.
- Debate
- Group discussions
- Facing Interviews
- Resume preparation
- e-correspondence
Minimum Requirement

Computer aided multi media language lab with 30 systems with LAN facility.
Conventional Language Lab. with audio and video systems, speakers, headphones and a teacher console to accommodate 30 students.

Suggested Software:

Cambridge Advanced Learners’ Dictionary with exercises
The Rosetta Stone English Library
Clarity Pronunciation Power
Mastering English in Vocabulary, Grammar, Spellings, Composition
Dorling Kindersley series of Grammar, Punctuation, Composition etc.
Oxford Advanced Learner’s Compass, 7th Edition
Language in Use, Foundation Books Pvt Ltd
Learning to Speak English - 4 CDs
Microsoft Encarta
Murphy’s English Grammar, Cambridge
Time series of IQ Test, Brain-teasers, Aptitude Test etc.
English in Mind, Herbert Puchta and Jeff Stranks with Meredith Levy, Cambridge

Books Suggested for English lab :

1. Developing Communication Skills by Krishna Mohan & Meera Benerji (Macmillan)
2. Speaking English Effectively by Krishna Mohan & NP Singh (Macmillan)
7. Lingua TOEFL CBT Insider, by Dreamtech
8. TOEFL & GRE( KAPLAN, AARCO & BARRONS, USA, Cracking GRE by CLIFFS)
9. English Skills for Technical Students, WBSCTE with British Council, OL
Write assembly language programs to evaluate the expressions:
   i)  \( a = b + c - d \times e \)
   ii) \( z = x \times y + w - v \times u / k \)

a. Considering 8-bit, 16 bit and 32 bit binary numbers as \( b, c, d, e \).
b. Considering 2 digit, 4 digit and 8 digit BCD numbers.

Take the input in consecutive memory locations and results also.
Display the results by using “int xx” of 8086. Validate program for the boundary conditions.

Write assembly language programs to evaluate the expressions:
   i)  \( a = b + c - d \times e \)
   ii) \( z = x \times y + w - v \times u / k \)

Considering 8-bit, 16 bit and 32 bit binary numbers as \( b, c, d, e \).

Write an ALP of 8086 to add two exponential numbers which are in IEEE 754 notation. Display the results by using “int xx” of 8086. Validate program for the boundary conditions.

Write an ALP of 8086 to take \( N \) numbers as input. And do the following operations on them.
   Arrange in ascending and Descending order.
   Find max and minimum
   Find average

Considering 8-bit, 16 bit binary numbers and 2 digit, 4 digit and 8 digit BCD numbers. Display the results by using “int xx” of 8086. Validate program for the boundary conditions.

4. Write an ALP of 8086 to take a string of as input (in ‘C’ format) and do the following operations on it.
   a. Find the length
   b. Find it is Palindrome or not
   c. Find whether given string substring or not.
   d. Reverse a string
   e. Concatenate by taking another string

Display the results by using “int xx” of 8086.

5. Write the ALP to implement the above operations as procedures and call from the main procedure.

6. Write an ALP of 8086 to find the factorial of a given number as a procedure and call from the main program which display the result.

7. A computer uses RAM chips of 1024 X 1 capacity.
   a. How many chips are needed, and how should their address lines be connected to provide a memory capacity of 1024 bytes?
   b. How many chips are needed to provide a memory capacity of 16K bytes?
8 A computer uses RAM chips of 1024 X 1 capacity.
   a. How many chips are needed, and how should their address lines be connected to provide a memory capacity of 1024 bytes?
   b. How many chips are needed to provide a memory capacity of 16K bytes?

9 A computer employs RAM chips of 256X8 and ROM chips of 1024 X 8. The computer needs 2K bytes of RAM, 4K bytes of ROM, and four interface units, each with four registers. A memory-mapped I/O configuration is used. The two highest-order bits of the address bus are assigned 00 for RAM, 01 for ROM, 10 for interface registers.
   a. How many RAM and ROM chips are needed?
   b. Draw a memory-address map for the system.
   c. Give the address range in hexadecimal for RAM, ROM and interface.

10 Obtain the complement function for the match logic of one word in an associative memory. Draw the logic diagram for it and compare with the actual match logic diagram.

11 A two-way set associative cache memory uses blocks of four words. The cache can accommodate a total of 2048 words from main memory. The main memory size is 128K X 32.
   a. Formulate all pertinent information required to construct the cache memory.
   b. What is the size of the cache memory?

12 A digital computer has a memory unit of 64K X 16 and a cache memory of 1K words. The cache uses direct mapping with a block size of four words.
   a. How many bits are there in each word of cache, and how are they divided into functions? Include a valid bit.
   b. How many bits are there in the tag, index, block, and word fields of the address format?
   c. How many blocks can the cache accommodate?

13 An address space is specified by 24 bits and the corresponding memory space by 16 bits.
   a. How many words are there in the address space?
   b. How many words are there in the memory space?
   c. If a page consists of 2K words, how many pages and blocks are there in the system.

14 A virtual memory has a page size of 1K words. There are eight pages and four blocks. The associative memory page table contains the following entries. Make a list of all virtual addresses(in decimal) that will cause a page fault.

<table>
<thead>
<tr>
<th>Page</th>
<th>Block</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
</tr>
</tbody>
</table>
1. Write a C++ program to find the sum of individual digits of a positive integer.

2. A Fibonacci sequence is defined as follows: the first and second terms in the sequence are 0 and 1. Subsequent terms are found by adding the preceding two terms in the sequence. Write a C++ program to generate the first n terms of the sequence.

3. Write a C++ program to generate all the prime numbers between 1 and n, where n is a value supplied by the user.

4. Write C++ programs that use both recursive and non-recursive functions
   To find the factorial of a given integer.
   To find the GCD of two given integers.
   To find the nth Fibonacci number.

5. Write a C++ program that uses a recursive function for solving Towers of Hanoi problem.

6. Write a C++ program that uses functions
   a) To swap two integers.
   b) To swap two characters.
   c) To swap two reals. Note: Use overloaded functions.

7. Write a C++ program to find both the largest and smallest number in a list of integers.

8. Write a C++ program to sort a list of numbers in ascending order.

9. Write a C++ program that uses function templates to solve problems-7&8.

10. Write a C++ program to sort a list of names in ascending order.

11. Write a C++ program to implement the matrix ADT using a class. The operations supported by this ADT are:
   a) Reading a matrix.  c) Addition of matrices.
   b) Printing a matrix.  d) Subtraction of matrices.
   e) Multiplication of matrices.

12. Implement the matrix ADT presented in the problem-11 using overloaded operators (<<, >>, +,-, *) and templates.
13. Implement the complex number ADT in C++ using a class. The complex ADT is used to represent complex numbers of the form $c=a+ib$, where $a$ and $b$ are real numbers. The operations supported by this ADT are:
   a) Reading a complex number.
   b) Writing a complex number.
   c) Addition of Complex numbers.
   d) Subtraction of complex numbers.
   e) Multiplication of complex numbers.
   f) Division of complex numbers.

14. Write a C++ program that overloads the + operator and relational operators (suitable) to perform the following operations:
   a) Concatenation of two strings.
   b) Comparison of two strings.

15. Implement the complex number ADT in C++ using a class. The complex ADT is used to represent complex numbers of the form $c=a+ib$, where $a$ and $b$ are real numbers. The operations supported by this ADT are:
   a) Reading a complex number.
   b) Writing a complex number.
   c) Addition of Complex numbers.
   d) Subtraction of complex numbers.
   e) Multiplication of complex numbers.
   f) Division of complex numbers.
   Note: 1. overload << and >> operators in part a and part b.
   2. overload +, -, *, / operators in parts c, d, e and f.

16. Write a template based C++ program that determines if a particular value occurs in an array of values.

17. Write a C++ program that uses functions to perform the following operations to:
   a) Insert a sub-string in to the given main string from a given position.
   b) Delete n characters from a given position in a given string.

18. Write a C++ program that uses a function to reverse the given character string in place, without any duplication of characters.

19. Write a C++ program to make the frequency count of letters in a given text.

20. Write a C++ program to count the lines, words and characters in a given text.

21. Write a C++ program to determine if the given string is a palindrome or not.

22. Write a C++ program to make frequency count of words in a given text.

23. Write a C++ program that displays the position or index in the string $S$ where the string $t$ begins, or $-1$ if $S$ doesn’t contain $t$.

24. 2’s complement of a number is obtained by scanning it from right to left and complementing all the bits after the first appearance of a 1. Thus 2’s complement of 11100 is 00100. Write a C++ program to find the 2’s complement of a binary number.
25. Write a C++ program that counts the number of 1 bit in a given integer.

26. Write a C++ program to generate Pascal’s triangle.

27. Write a C++ program to construct a pyramid of numbers.

28. Write a C++ program to compute the Sine series.

29. Write a C++ program that converts Roman numeral into an Arabic integer.

30. Write a C++ program which converts a positive Arabic integer into its corresponding Roman Numeral.

31. Write a C++ program to display the contents of a text file.

32. Write a C++ program which copies one file to another.

33. Write a C++ program that counts the characters, lines and words in the text file.

34. Write a C++ program to change a specific character in a file.
   Note: Filename, number of the byte in the file to be changed and the new character are specified on the command line.

35. Write a C++ program to reverse the first n characters in a file.

36. Write a C++ program that uses a function to delete all duplicate characters in the given string.

37. Write a C++ program that uses a function (i to a) which converts a number to a character string.

38. Write a C++ program that uses a recursive function to find the binary equivalent of a given non-negative integer n.

39. Write a C++ program to generate prime numbers up to n using Sieve of Eratosthenes method.

40. Write a C++ program
   a) To write an object to a file.
   b) To read an object from the file.
41. Write C++ programs that illustrate how the following forms of inheritance are supported:
   a) Single inheritance
   b) Multiple inheritance
   c) Multi level inheritance
   d) Hierarchical inheritance

42. Write a C++ program that illustrates the order of execution of constructors and destructors when new class is derived from more than one base class.

43. Write a C++ program that illustrates how run time polymorphism is achieved using virtual functions.

44. Write a C++ program that illustrates the role of virtual base class in building class hierarchy.

45. Write a C++ program that illustrates the role of abstract class in building class hierarchy.

46. Write a C++ program that uses functions:
   a) To create a singly linked list of elements
   b) To display the elements of the above list.

**Note:** Use the following in solving the above problems wherever they make sense:
   a) Constructors and destructors.
   b) Overloaded functions.
   c) Overloaded operators.
   d) Function and class templates.
   e) Exception handling mechanism.

**Suggested Books for lab:**
MC 2.1 DATA STRUCTURES (THROUGH JAVA)

UNIT-I
Java Basics: Creation of java, java buzzwords, data types, variables, operators, expressions, control flow statements- if and switch statements, loops- for, while, do-while, jump statements, classes, objects, class modifiers, class members and access control, methods, simple input and output statements, an example program, string handling, inheritance- super and sub classes, member access rules, method overriding, dynamic method dispatch, abstract classes, Object class, interfaces- implementing interfaces, multiple inheritance in interfaces.

UNIT-II
Packages, Exception Handling And Multi Threading: Packages- defining, creating and accessing a package, importing packages, exception handling- concepts of exception handling, types of exceptions, usage of try, catch, throw, throws and finally key words, creating own exception sub classes, multi threading- thread life cycle, creating multiple threads using Thread class, Runnable interface, java library- java.util, java.io.

UNIT-III
Introduction To Algorithms And Data Structures: Notion of algorithm, pseudo code, performance analysis- time complexity and space complexity, asymptotic notation (big-oh, omega, theta), data abstraction, concept of ADT, linear and non linear data structures, sequential and linked allocation, arrays and linked lists, representation of single, two and multi dimensional arrays, sparse matrices and their representation, the list ADT-array based implementation and linked list implementation.

UNIT-IV
Stacks And Queues: Stack ADT-array based implementation, linked list implementation, applications-infix to postfix conversion, postfix evaluation, implementation of recursion, Queue ADT- array based implementation, linked list implementation, circular queues, Dequeue ADT- array based implementation, linked list implementation, Priority Queues ADT- implementation, heaps.
UNIT-V

Trees And Graphs: Trees- Terminology and basic properties, tree ADT, binary tree ADT, data structures for representing binary trees - a vector based structure and linked structure, traversals of a binary tree, representing general trees with binary trees, threaded binary trees, graphs- graph ADT, basic terminology, data structures for representing graphs- edge list structures, adjacency list structures, adjacency matrix, graph traversals- DFS, BFS.

UNIT-VI

Search Trees: Binary search trees, operations- insertion, deletion and searching, AVL trees, operations- insertion and searching, B-trees, operations- insertion, deletion and searching, comparison of performance of search trees.

UNIT-VII

Searching And Sorting: Linear search, binary search, hashing- hash table, its implementation, hash functions, collision handling schemes, bubble sort, selection sort, insertion sort, quick sort, merge sort, heap sort, radix sort, comparison of searching and sorting methods.

UNIT-VIII

Text Processing: Pattern matching algorithms- Brute force, the Boyer-Moore algorithm, the KMP algorithm, tries- standard tries, compressed tries, suffix tries, text compression- Huffman coding algorithm.

TEXT BOOKS:


Reference Books:


3. Data Structures with Java, J.R.Hubbard and A.Huray, Pearson Education/PHI.


5. Big Java, C. Horstmann, John Wiley.
MC 2.2 OPERATING SYSTEMS

UNIT I:

UNIT II:
Process and CPU Scheduling - Process concepts and scheduling, Operation on processes, Cooperating Processes, Threads, and Interposes Communication Scheduling Criteria, Scheduling Algorithm, Multiple-Processor Scheduling, Real-Time Scheduling.

UNIT III

UNIT IV:

UNIT V:

UNIT VI:
Deadlocks - System Model, Deadlocks Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, and Recovery from Deadlock.

UNIT VII:
UNIT VIII:

TEXT BOOKS:


REFERENCE BOOKS:

1. Operating System A Design Approach-Crowley, TMH.
3. Operating Systems, Dhamdhere, TMH
MC 2.3 ORGANIZATION STRUCTURE AND PERSONAL MANAGEMENT

UNIT-I
Classical Theories of organization: Functional approach, classical theories of organization, division of labour, levels of authority, span of control, authority & responsibility, efficiency of management.

UNIT-II
Behavioural theories of organization, limitations of formal organization, human relation, group behaviour, committee and group making, motivation and morale.

UNIT-III
Decision process approach: Parts of organization system, development of corporate strategy, dynamics of decision, role of system, types models, mathematical planning models, deterministic and probabilistic models.

UNIT-IV
Personnel Function: Evaluation, objectives, principles, philosophies and policies, duties & responsibilities of the manager, position of the personnel department in the organization, line and staff relationship & the changing concept of personnel management in India.

UNIT-V
Manpower planning: Uses benefits problems and limitations, manpower, inventory, manpower forecasting, job discription, recruitment, job specification and job selection interviewing techniques, transfers, promotion and its policies.

UNIT-VI
Training and development: Objectives and policies planning, organizing the training department, training manager and his job, on and off the job training, techniques, career planning, objectives of performance appraisal.

UNIT-VII
Strategic management: Objectives, importance policies, concept of core competence capability of organizational learning.
UNIT-VIII
Communication: Importance of communication, interpersonal communication barriers of communication, communication in organizations, using communication skills to manage conflicts.

Text Book:
MC 2.4 OPERATIONS RESEARCH

UNIT-I
ALLOCATION: Introduction, Linear programming Formulation, Graphical solution, Simplex method, artificial variable technique, Duality principle.

UNIT-II

UNIT-III
EEQUE CEING: Introduction, optimal solution for processing each of n-jobs through three machines, travelling salesman problem i.e., shortest acyclic route models.

UNIT-IV
REPLACEMENT: Introduction, replacement of items that deteriorate when money value is not counted and counted, replacement items that fail completely i.e., group replacements.

UNIT-V
WAITING LINES: Introduction, single channel, poisson arrivals, exponential service times, unrestricted queue, with infinite population and finite population models, single channel, poisson arrivals, exponential service times with infinite population and restricted queue, multi channel, poisson arrivals, exponential service times with infinite population and unrestricted queue.

UNIT-VI
INVENTORY: Introduction, single item deterministic models, production is instantaneous or at a constant rate, shortages are allowed or not allowed and withdrawals from stock is continuous, purchase inventory model with one price break, shortages are not allowed, Instantaneous production demand, production or purchase cost is relevant, stochastic models, demand may be discrete or variable or instantaneous production, instantaneous demand and no setup cost.

UNIT-VII
THEORY OF GAMES: Introduction, Minimax (maximum) criterion and optimal strategy, solution of games with saddle points, rectangular games without saddle points.
UNIT-VIII
DYNAMIC PROGRAMMING: Introduction, Billman’s Principal of optimality, solution of problems with finite number of stages.

TEXT BOOKS:
1. S.D.SHARMA : Operations Research
MCA 2.5 BUSINESS DATA PROCESSING

UNIT - I:
The IDENTIFICATION and ENVIRONMENT DIVISIONS: Basic Structure of a COBOL program, Coding Requirements of the Identification Division, The Sections of the Environment division, Assigning files to Devices in the Environment Division.

UNIT - II
The DATA DIVISION: Systems Design considerations, Forming Data-Names, The FILE SECTION of the Data Division, Types of Data, The working-Storage section of the Data division.
Coding Complete COBOL Programs: The PROCEDURE DIVISION, The format of the Procedure division, Statements typically coded in the Main Module of Batch Programs, Statements typically coded for Processing Input records and Producing output records.

UNIT - III
Moving Data, Printing Information, and Displaying Output Interactively, The instruction formats of the MOVE STATEMENT, Numeric MOVE, Nonnumeric or Alphanumeric MOVE, Other Options of the MOVE STATEMENT. PRINTING OUTPUT, Interactive output that is displayed on a screen.

UNIT - IV
Decision Making Using the IF and EVALUATE Statements, Selection using a simple IF statement, Selection using other Options of the IF statement, CONDITION-NAMES.
Iteration: The simple PERFORM, Iteration using other types of PERFORMs, Using Nested PERFORM varying statements.

UNIT - V
Control Break Processing: An introduction to control break processing, Program Requirements for control break processing, Multiple-level control breaks.
Data Validation: Avoiding logic errors by validating input, What to do if input errors occur, When data should be validated, Understanding program interrupts.

UNIT - VI
Single level OCCURS clause, Processing data stored in an array, Using an OCCURS clause for Table Handling, Use of the Search statement for Table and Array processing, Varying option of SEARCH verb
SEARCH ALL statement, Multiple level OCCURS Clause.
Systems overview of Sequential processing: Sequential file updating, Validity checking in update procedures, Update procedures with multiple transaction records, Rewriting records on a disk.
UNIT - VII
Sorting and Merging: The SORT features - an overview, Processing data before/after sorting, MERGE statement,
Indexed and Relative File Processing: Systems considerations for organizing disk files, Features of magnetic disks and disk drives, Processing indexed disk files, Processing relative Disk Files, Converting a key field to a relative key.

UNIT - VIII
Program performance: The COPY, CALL and other Statements, COPY statement, CALL statement, Text Manipulation with the STRING and UNSTRING STATEMENT.

Text Book:
Cobol Programming – Collopy – Pearson Ed.
Nancy Stern and Robert A.Stern - Structured COBOL programming 8th Ed. - John Wiley & Sons
1. Write a Java program that prints all real solutions to the quadratic equation $ax^2 + bx + c = 0$. Read in $a$, $b$, $c$ and use the quadratic formula. If the discriminant $b^2 - 4ac$ is negative; display a message stating that there are no real solutions.

2. The Fibonacci sequence is defined by the following rule. The first two values in the sequence are 1 and 1. Every subsequent value is the sum of the two values preceding it. Write a Java program that uses both recursive and non-recursive functions to print the $n$th value in the Fibonacci sequence.

3. Write a Java program that prompts the user for an integer and then prints out all prime numbers up to that integer.

4. Write a Java program that checks whether a given string is a palindrome or not. Ex: MADAM is a palindrome.

5. Write a Java program for sorting a given list of names in ascending order.

6. Write a Java program to multiply two given matrices.

7. Write a Java Program that reads a line of integers, and then displays each integer, and the sum of all the integers (use StringTokenizer class).

8. Write a Java program that reads a file name from the user then displays information about whether the file exists, whether the file is readable, whether the file is writable, the type of file and the length of the file in bytes.

9. Write a Java program that reads a file and displays a file and displays the file on the screen, with a line number before each line.

10. Write a Java program that displays the number of characters, lines and words in a text file.

11. Write a Java program for creating multiple threads
   a) Using Thread class
   b) Using Runnable interface

12. Write a Java program that illustrates how run time polymorphism is achieved.
13. Write a java program that illustrates the following
   a) Creation of simple package.
   b) Accessing a package.
   c) Implementing interfaces.

14. Write a java program that illustrates the following
   a) Handling predefined exceptions
   b) Handling user defined exceptions

15. Write Java programs that use both recursive and non-recursive functions for implementing the following searching methods:
   a) Linear search
   b) Binary search

16. Write java programs to implement the following using arrays and linked lists
   d) List ADT
   a) Write Java programs to implement the following using an array.
      a) Stack ADT
      b) Queue ADT

18. Write a java program that reads an infix expression, converts the expression to postfix form and then evaluates the postfix expression (use stack ADT).

19. Write a java program that determines whether parenthetic symbols ( ), { } and < > are nested correctly in a string of characters (use stack ADT).

20. Write a java program that uses both stack and queue to test whether the given string is a palindrome.

21. Write Java programs to implement the following using a singly linked list.
   a) Stack ADT
   b) Queue ADT

22. Write Java programs to implement the deque (double ended queue) ADT using
   a) Array
   b) Singly linked list
   c) Doubly linked list.

23. Write a java program to implement priority queue ADT.

24. Write a Java program to perform the following operations:
   a) Insert an element into a binary search tree.
   b) Delete an element from a binary search tree.
   c) Search for a key element in a binary search tree.
25. Write a Java program to implement all the functions of a dictionary (ADT) using Hashing.

26. Write a Java program to implement circular queue ADT using an array.

27. Write Java programs that use recursive and non-recursive functions to traverse the given binary tree in
   a) Preorder
   b) Inorder and
   c) Postorder.

28. Write Java programs for the implementation of bfs and dfs for a given graph.

29. Write Java programs for implementing the following sorting methods:
   a) Bubble sort  d) Quick sort  g) Radix sort
   b) Selection sort  e) Merge sort  h) Binary tree sort
   c) Insertion sort  f) Heap sort

30. Write a Java program to perform the following operations
    a) Insertion into a B-tree
    b) Deletion from a B-tree

31. Write a Java program to perform the following operations
    a) Insertion into an AVL-tree
    b) Deletion from an AVL-tree

32. Write a Java program for implementing KMP pattern matching algorithm.

33. Write a java program that displays node values in a level order traversal (traverse the tree one level at a time, starting at the root node) for a binary tree.

34. Write a java program that uses recursive functions
   a. To create a binary search tree.
   b. To count the number of leaf nodes.
   c. To copy the above binary search tree.

35. Write a Java program to implement all the functions of a dictionary (ADT) using Hashing.

**Suggested Books for lab:**
MC 3.1 DATA BASE MANAGEMENT SYSTEMS

UNIT – I:

UNIT – II:

UNIT – III:
Form of Basic SQL Query – Examples of Basic SQL Queries – Introduction to Nested Queries – Correlated Nested Queries Set – Comparison Operators – Aggregative Operators – NULL values – Comparison using Null values – Logical connectivity’s – AND, OR and NOTR – Impact on SQL Constructs – Outer Joins – Disallowing NULL values – Complex Integrity Constraints in SQL Triggers and Active Data bases.

UNIT – IV:

UNIT – V:
UNIT – VI:


UNIT – VII:


UNIT – VIII:


TEXT BOOKS:


REFERENCE BOOK:

1. Introduction to Database Systems, C.J.Date Pearson Education
3. Data base Management System, Elmasri Navrate Pearson Education
5. Data base Systems, Connoley Pearson education
UNIT-I

UNIT-II
Physical Layer: Guided transmission media – Magnetic media, Twisted Pair, coaxial cable, fiber optics.


UNIT-III
The Medium Access Sub Layer: The channel allocation problem, Multiple access Protocols, Ethernet, Wireless LANs, Broadband Wireless, Bluetooth, Data Link Layer Switching.

UNIT-IV

UNIT-V

UNIT-VI
The Application Layer: DNS-(Domain Name System), Electronic Mail, World Wide Web Multimedia,

UNIT-VII

UNIT-VIII
TEXT BOOKS:


REFERENCE BOOKS:


UNIT-I
Unix Utilities-Introduction to Unix file system, vi editor, file handling utilities, security by file permissions, process utilities, disk utilities, networking commands, cp, mv, ln, rm, unlink, mkdir, rmdir, du, df, mount, umount, find, unmask, ulimit, ps, who, w, finger, arp, ftp, telnet, rlogin, text processing utilities and backup utilities, detailed commands to be covered are cat, tail, head, sort, nl, uniq, grep, egrep, fgrep, cut, paste, join, tee, pg, comm, cmp, diff, tr, awk, tar, cpio.

UNIT-II
Problem solving approaches in Unix: Using single commands, using compound Commands, shell scripts, C programs, building own command library of programs.
Working with the Bourne shell: what is a shell, shell responsibilities, pipes and input Redirection, output redirection, here documents, the shell as a programming language, shell meta characters, shell variables, shell commands, the environment, control structures, shell script examples.

UNIT-III
Unix Files: Unix file structure, directories, files and devices, System calls, library functions, low level file access, usage of open, creat, read, write, close, lseek, stat, fstat, octl, umask, dup, dup2.
The standard I/O (fopen, fclose, fflush, fseek, fgets, getc, getchar, fputc, putc, putchar, fgets, gets ), formatted I/O, stream errors, streams and file descriptors, file and directory maintenance (chmod, chown, unlink, link, symlink, mkdir, rmdir, chdir, getcwd), Directory handling system calls (opendir, readdir, closedir, rewinddir, seekdir, telldir)

UNIT-IV
Unix Process and Signals: What is process, process structure, starting new process, waiting for a process, zombie process, process control, process identifiers, system call interface for process management-fork, vfork, exit, wait, waitpid, exec, system, Signals- Signal functions, unreliable signals, interrupted system calls, kill and raise functions, alarm, pause functions, abort, sleep functions.

UNIT-V
Interprocess Communication Overview: Introduction to IPC,IPC between processes on a single computer system, IPC between processes on different systems, file and record locking, other unix locking techniques, pipes, FIFOs, streams and messages, namespaces, introduction to three types of IPC(system-V)-message queues, semaphores and shared memory.
UNIT-VI
Message Queues—Unix system-V messages, unix kernel support for messages, unix APIs for messages, client/server example.

UNIT-VII
Semaphores—Unix system-V semaphores, unix kernel support for semaphores, unix APIs for semaphores, file locking with semaphores.
Shared Memory—Unix system-V shared memory, unix kernel support for shared memory, unix APIs for shared memory, semaphore and shared memory example.

UNIT-VIII
Sockets: Berkeley sockets, socket system calls for connection oriented protocol and connectionless protocol, example-client/server program.

TEXT BOOKS:
1. Unix Network Programming, W.R.Stevens Pearson/PHI.

REFERENCE BOOKS:
1. Unix system programming using C++, T.Chan, PHI.
4. Unix programming environment, Kernighan and Pike, PHI. / Pearson Education
UNIT I:
The meaning and role of MIS: What is MIS?, Decision support systems, systems approach, the systems view of business, MIS organization within the company, Managers view of Information systems.

UNIT II:
Management organizational theory and the systems approach Development of organizational theory, management and organizational behavior, management, information, and the systems approach, using Information systems for feedback.

UNIT III:
Information systems for decision making: Evolution of an information system, Basic information systems, decision making and MIS, MIS as a technique for making programmed decisions, decision assisting information systems. Communication systems basics.

UNIT IV:
Strategic and project planning for MIS: General business planning, appropriate MIS response, MIS planning-general, MIS planning-details.

UNIT V:
Conceptual system design: Define the problems, set system objectives, establish system constraints, determine information needs, determine information sources, develop alternative conceptual designs and select one, document the system concept, prepare the conceptual design report.

UNIT VI:
Detailed system design: Inform and involve the organization, aim of detailed design, project management of MIS detailed design, identify dominant and trade off criteria, define the subsystems, sketch the detailed operating subsystems and information flows, determine the degree of automation of each operation, inform and involve the organization again, inputs, outputs, and processing, early system testing, software, hardware and tools, propose an organization to operate the system, document the detailed design, revisit the manager-user.
UNIT VII:
Implementation, evaluation and maintenance of the MIS: Plan the implementation, acquire floor space and plan space layouts, organize for implementation, develop procedures for implementation, train the operating personnel, computer related acquisitions, develop forms for data collection and information dissemination, develop the files, test the system, cutover, document the system, evaluate the MIS, control and maintain the system.

UNIT VIII:
Pitfalls in MIS development: Fundamental weaknesses, soft spots, in planning, design problems, implementation: the TAR PIT. Applications of information systems to business. Security and ethical issues of information systems.

TEXT BOOK:


REFERENCE BOOK:

2. Management Information Systems, 9/e, Laudon & Laudon, V.M.Prasad, Pearson, 2005,
MC 3.5 COMPUTER GRAPHICS

UNIT-I:

UNIT-II:

UNIT-III:
2-D geometrical transforms: Translation, scaling, rotation, reflection and shear transformations, matrix representations and homogeneous coordinates, composite transforms, transformations between coordinate systems (p.nos 204-227 of text book-1).

UNIT-IV:

UNIT-V:

UNIT-VI:
3-D Geometric transformations: Translation, rotation, scaling, reflection and shear transformations, composite transformations. 3-D viewing: Viewing pipeline, viewing coordinates, view volume and general projection transforms and clipping (p.nos 427-443, 452-481 of text book-1).
UNIT-VII:
Visible surface detection methods: Classification, back-face detection, depth-buffer, scan-line, depth sorting, BSP-tree methods, area sub-division and octree methods(p.nos 489-505 of text book -1, Chapter 15 of text book-2).

UNIT-VIII:

TEXT BOOKS:


REFERENCE BOOKS:


6. Computer Graphics, Steven Harrington, TMH
1. Creating tables for various relations (in SQL)

2. Implementing the queries in SQL for
   a) Insertion
   b) Retrieval (Implement all the operation like Union, Intersect, Minus, in, exist, aggregate functions (Min., Max…) etc…
   c) Updation
   d) Deletion

3. Creating Views

4. Writing Assertions

5. Writing Triggers

6. Implementing Operations on relations (tables) using PL/SQL

7. Creating FORMS

8. Generating REPORTS.
MC 3.7 UNIX NETWORK PROGRAMMING LAB

1. Write a shell script to generate a multiplication table.

2. Write a shell script that copies multiple files to a directory.

3. Write a shell script that counts the number of lines and words present in a given file.

4. Write a shell script that displays the list of all files in the given directory.

5. Write a shell script (small calculator) that adds, subtracts, multiplies and divides the given two integers. There are two division options: one returns the quotient and the other returns reminder. The script requires 3 arguments: The operation to be used and two integer numbers. The options are add (-a), subtract (-s), multiply (-m), quotient (-c) and reminder (-r).

6. Write a shell script to reverse the rows and columns of a matrix.

7. Write a C program that counts the number of blanks in a text file.
   Using standard I/O
   Using system calls.

8. Implement in C the following Unix commands using system calls.
   a) cat
   b) ls
   mv

9. Write a program that takes one or more file/directory names as command line input and reports the following information on the file:
   a) File type.
   b) Number of links.
   c) Time of last access.
   d) Read, Write and Execute permissions.

10. Write a C program that illustrates uses of the mkdir, opendir, readdir, closedir, and rmdir APIs.

11. Write a C program that illustrates how to execute two commands concurrently with a command pipe.
12. Write a C program that illustrates the following:
   a) Two-way communication with unidirectional pipes.
   b) Two-way communication with bidirectional pipes

13. Write a C program that illustrates the creation of child process using fork system call.

14. Write a C program that displays the real time of a day every 60 seconds.

15. Write a C program that illustrates file-locking using semaphores.

16. Write a C program that implements a producer-consumer system with two processes. (Using semaphores)

17. Write a C program that illustrates inter process communication using shared memory system calls.

18. Write a C program that illustrates the following.
    a) Creating a message queue.
    b) Writing to a message queue.
    b) Reading from a message queue.

19. Write a C program to develop simple client and server application using sockets (system calls).

Suggested textbooks for lab:

UNIT-I:

UNIT-II:
Process models: The waterfall model, Incremental process models, Evolutionary process models, The Unified process.
Software Requirements: Functional and non-functional requirements, User requirements, System requirements, Interface specification, the software requirements document.

UNIT-III:
Requirements engineering process: Feasibility studies, Requirements elicitation and analysis, Requirements validation, Requirements management.
System models: Context Models, Behavioral models, Data models, Object models, structured methods.

UNIT-IV:
Design Engineering: Design process and Design quality, Design concepts, the design model.
Creating an architectural design: software architecture, Data design, Architectural styles and patterns, Architectural Design.

UNIT-V:
Object-Oriented Design: Objects and object classes, An Object-Oriented design process, Design evolution.
Performing User interface design: Golden rules, User interface analysis and design, interface analysis, interface design steps, Design evaluation.

UNIT-VI:
Testing Strategies: A strategic approach to software testing, test strategies for conventional software, Black-Box and White-Box testing, Validation testing, System testing, the art of Debugging.
Product metrics: Software Quality, Metrics for Analysis Model, Metrics for Design Model, Metrics for source code, Metrics for testing, Metrics for maintenance.
UNIT-VII:
**Metrics for Process and Products:** Software Measurement, Metrics for software quality.
**Risk management:** Reactive vs Proactive Risk strategies, software risks, Risk identification, Risk projection, Risk refinement, RMMM, RMMM Plan.

UNIT-VIII:
**Quality Management:** Quality concepts, Software quality assurance, Software Reviews, Formal technical reviews, Statistical Software quality Assurance, Software reliability, The ISO 9000 quality standards.

TEXT BOOKS:


REFERENCE BOOKS:

MC 4.2 ADVANCED JAVA PROGRAMMING

UNIT-I:
HTML Common tags - List, Tables, images, forms, Frames; Cascading Style sheets; Introduction to Java Scripts, Objects in Java Script, Dynamic HTML with Java Script

UNIT -II:

UNIT- III:
Review of Applets, Class, Event Handling, AWT Programming
Introduction to Swing:

UNIT - IV:
Java Beans Introduction to Java Beans, Advantages of Java Beans, BDK Introspection, Using Bound properties, Bean Info Interface, Constrained properties Persistence, Customizers, Java Beans API

UNIT - V:
Introduction to Servelets: Lifecycle of a Serverlet, JSDK The Servelet API, The javax.servlet Package, Reading Servelet parameters, Reading Initialization parameters. The javax.servlet HTTP package, Handling Http Request & Responses, Using Cookies-Session Tracking, Security Issues

UNIT - VI:
Introduction to JSP The Problem with Servelet. The Anatomy of a JSP Page, JSP Processing.
JSP Application Design with MVC Setting Up and JSP Environment: Installing the Java Software Development Kit, Tomcat Server & Testing Tomcat

UNIT - VII:
JSP Application Development: Generating Dynamic Content, Using Scripting Elements Implicit JSP Objects, Conditional Processing – Displaying Values Using an Expression to Set an Attribute, Declaring Variables and Methods Error Handling and Debugging Sharing Data Between JSP pages, Requests, and Users Passing Control and Date between Pages – Sharing Session and Application Data – Memory Usage Considerations
UNIT - VIII:
Database Access Database Programming using JDBC Studying Javax.sql.* package
Accessing a Database from a JSP Page Application – Specific Database Actions
Deploying JAVA Beans in a JSP Page. Introduction to struts framework.

TEXT BOOKS:
Internet and World Wide Web – How to program by Dietel and Nieto Pearson
   Education Asia. (Chapters: 3, 4, 8, 9, 10, 11, 12 – 18)
The complete Reference Java 2 Third Edition by Patrick Naughton and Herbert
   Schiltdt. (Chapters: 19, 20, 21, 22, 25, 27)
Java Server Pages by Hans Bergstan. (Chapters: 1 – 9)

REFERENCE BOOKS:
1. Internet and World Wide Web – How to program by Dietel and Nieto PHI/Pearson
   Education Asia.
3. Murach’s beginning JAVA JDK 5, Murach, SPD
5. Web Applications Technologies Concepts-Knuckles,John Wiley
6. Programming world wide web-Sebesta,Pearson
7. Building Web Applications-NIIT,PHI
8. Web Warrior Guide to Web Programmming-Bai/Ekedaw-Thomas
9. Beginning Web Programming-Jon Duckett WROX.
MC 4.3 DATA WAREHOUSING AND MINING

UNIT-I
Introduction: Fundamentals of data mining, Data Mining Functionalities, Classification of Data Mining systems, Major issues in Data Mining, Data Warehouse and OLAP Technology for Data Mining Data Warehouse, Multidimensional Data Model, Data Warehouse Architecture, Data Warehouse Implementation, Further Development of Data Cube Technology, From Data Warehousing to Data Mining,

UNIT-II
Data Preprocessing: Needs Preprocessing the Data, Data Cleaning, Data Integration and Transformation, Data Reduction, Discretization and Concept Hierarchy Generation, Online Data Storage.

UNIT-III
Data Mining Primitives, Languages, and System Architectures: Data Mining Primitives, Data Mining Query Languages, Designing Graphical User Interfaces Based on a Data Mining Query Language Architectures of Data Mining Systems,

UNIT-IV
Concepts Description: Characterization and Comparison: Data Generalization and Summarization-Based Characterization, Analytical Characterization: Analysis of Attribute Relevance, Mining Class Comparisons: Discriminating between Different Classes, Mining Descriptive Statistical Measures in Large Databases.

UNIT-V
Mining Association Rules in Large Databases: Association Rule Mining, Mining Single-Dimensional Boolean Association Rules from Transactional Databases, Mining Multilevel Association Rules from Transaction Databases, Mining Multidimensional Association Rules from Relational Databases and Data Warehouses, From Association Mining to Correlation Analysis, Constraint-Based Association Mining.

UNIT-VI
Classification and Prediction: Issues Regarding Classification and Prediction, Classification by Decision Tree Induction, Bayesian Classification, Classification by Back propagation, Classification Based on Concepts from Association Rule Mining, Other Classification Methods, Prediction, Classifier Accuracy.

UNIT-VII
Cluster Analysis Introduction: Types of Data in Cluster Analysis, A Categorization of Major Clustering Methods, Partitioning Methods, Density-Based Methods, Grid-Based Methods, Model-Based Clustering Methods, Outlier Analysis.
UNIT-VIII

**Mining Complex Types of Data:** Multidimensional Analysis and Descriptive Mining of Complex, Data Objects, Mining Spatial Databases, Mining Multimedia Databases, Mining Time-Series and Sequence Data, Mining Text Databases, Mining the World Wide Web.

**TEXT BOOKS:**

1. Data Mining – Concepts and Techniques - JIAWEI HAN & MICHELLE KAMBER Harcourt India.
2. Data Mining Techniques – ARUN K PUJARI, University Press
3. Building the DataWarehouse- W. H. Inmon, Wiley Dreamtech India Pvt. Ltd..

**REFERENCE BOOKS:**

2. Data Warehousing Fundamentals – PAULRAJ PONNAIAH WILEY STUDENT EDITION
3. The Data Warehouse Life cycle Tool kit – RALPH KIMBALL WILEY STUDENT EDITION
4. Data Mining Introductory and advanced topics – MARGARET H DUNHAM, PEARSON EDUCATION
MC 4.4.1 EMBEDDED SYSTEMS (ELECTIVE-I)

UNIT - I

UNIT - II
The 8051 Architecture: Introduction, 8051 Microcontroller Hardware, Input/Output Ports and Circuits, External Memory, Counter and Timers, Serial data Input/Output, Interrupts. (Chapter 3 from Text Book 2, Ayala)

UNIT - III
Basic Assembly Language Programming Concepts: The Assembly Language Programming Process, Programming Tools and Techniques, Programming the 8051, Data Transfer and Logical Instructions. (Chapters 4, 5 and 6 from Text Book 2, Ayala)

UNIT - IV
Arithmetic Operations, Decimal Arithmetic, Jump and Call Instructions, Further Details on Interrupts. (Chapter 7 and 8 from Text Book 2, Ayala)

UNIT - V
Applications: Interfacing with Keyboards, Displays, D/A and A/D Conversions, Multiple Interrupts, Serial Data Communication. (Chapter 10 and 11 from Text Book 2, Ayala)

UNIT - VI
Introduction to Real – Time Operating Systems: Tasks and Task States, Tasks and Data, Semaphores, and Shared Data; Message Queues, Mailboxes and Pipes, Timer Functions, Events, Memory Management, Interrupt Routines in an RTOS Environment (Chapter 6 and 7 from Text Book 3, Simon)

UNIT - VII
Basic Design Using a Real-Time Operating System: Principles, Semaphores and Queues, Hard Real-Time Scheduling Considerations, Saving Memory and Power, An example RTOS like uC-OS (Open Source); Embedded Software Development Tools: Host and Target machines, Linker/Locators for Embedded Software, Getting Embedded Software into the Target System; Debugging Techniques: Testing on Host Machine, Using Laboratory Tools, An Example System. (Chapter 8, 9, 10 and 11 from Text Book 3, Simon)
UNIT - VIII
Introduction to advanced architectures: ARM and SHARC, Processor and memory organization and Instruction level parallelism; Networked embedded systems: Bus protocols, I²C bus and CAN bus; Internet-Enabled Systems, Design Example-Elevator Controller.
(Chapter 8 from Text Book 1, Wolf)

Text Books:

Computers and Components, Wayne Wolf, Elseveir.
An Embedded Software Primer, David E. Simon, Pearson Education.

Reference Books:

Embedding system building blocks, Labrosse, via CMP publishers.
Embedded Systems, Raj Kamal, TMH.
Micro Controllers, Ajay V Deshmukhi, TMH.
Microcontrollers, Raj kamal, Pearson Education.
MC 4.4.2 HUMAN COMPUTER INTERACTION (ELECTIVE I)

UNIT-I
Introduction: Importance of user Interface – definition, importance of good design. Benefits of good design. A brief history of Screen design,

UNIT-II
The graphical user interface – popularity of graphics, the concept of direct manipulation, graphical system, Characteristics, Web user – Interface popularity, characteristics-Principles of user interface.

UNIT-III
Design process – Human interaction with computers, importance of human characteristics human consideration, Human interaction speeds, understanding business junctions.

UNIT-IV

UNIT-V

UNIT-VI
Components – text and messages, Icons and increases – Multimedia, colors, uses problems, choosing colors.

UNIT-VII

UNIT-VIII
TEXT BOOKS:

The essential guide to user interface design, Wilbert O Galitz, Wiley DreamTech.

Designing the user interface. 3rd Edition Ben Shneidermann, Pearson Education Asia

REFERENCE BOOKS:


2. Interaction Design Prece, Rogers, Sharps. Wiley Dreamtech,

MC 4.4.3 PERL PROGRAMMING

UNIT-I
Introduction to perl: # ! , Basic I/O, Variables variables & Backslash Interpolation, Scalar/list control operators, operator procedure, if unless, loops, loop control (ch1, ch3, pg 60-65, 69-72, III-127)

UNIT-II
Debugging perl scripts: Debugging commands, Debugger customization, Unattended execution, Debugging support, the perl profiler.

UNIT-III
Built in Function: Perl functions by category, Perl functions in alphabetical order.

UNIT-IV
Regular expressions: Pattern Matching, operators, Meta character and meta symbols. Character classes, quantifiers, Pointers, capturing & clustering, Alternation, staying in control.

UNIT-V
Subroutines Syntax: Syntax, Semantics, parsing references, prototypes, subroutine attributes.
Formats: Format variables, Fosters.

UNIT-VI
References: Creating References, using hard references, symbolic references, Braces, Brackets and quotes.
Data Structure: Arrays of Arrays, Hashes of arrays, Arrays of Hashes, Hashes of Hashes, Hashing as function, Elaborate records, Hashes of functions.
UNIT-VII
CGI Programming: CGI Basic, Forms, Methods.

UNIT-VIII
More CGI: Here Docs, Emailing, Cookies, File uploading, E-mail.

Text Books:

2. CGI Programming with Perl Scott Guelich, et al., O’Reily
MC 4.5.1 - DISTRIBUTED OPERATING SYSTEMS (Elective – II)

UNIT-I
Introduction to Distributed Systems:
Distributed systems: Goals Hardware Concepts Software - design

UNIT-II
Communication distributed systems:

UNIT-III
Synchronization:
Clock synchronization - mutual exclusion - election atomic transactions - dead locks.

UNIT-IV
Process and Processors:
Threads - System models processor allocation - scheduling fault tolerance - real time distributed systems.

UNIT-V
Distributed file systems:
File system design and implementation - trends in distributed file systems.

UNIT-VI
Shared Memory:
Introduction - bus based multi processors ring based multiprocessors switched multiprocessors - NUMA comparison of shared memory systems -

UNIT-VII
consistency models - page based distributed shared memory - shared variable distributed shared memory - object based distributed shared memory.

UNIT-VIII
Case studies: MACH and CHORUS

Text Book:
MC 4.5.2 MOBILE COMPUTING(Elective – II)

UNIT – I Introduction to Network Technologies and Cellular Communications:
HIPERLAN: Protocol architecture, physical layer, Channel access control sub-layer, MAC sub-layer, Information bases and networking
WLAN: Infrared vs. radio transmission, Infrastructure and ad hoc networks, IEEE 802.11. Bluetooth.: User scenarios, Physical layer, MAC layer, Networking, Security, Link management
GSM: Mobile services, System architecture, Radio interface, Protocols, Localization and calling, Handover, Security, and New data services.
Mobile Computing (MC): Introduction to MC, novel applications, limitations, and architecture

UNIT –II (Wireless) Medium Access Control: Motivation for a specialized MAC (Hidden and exposed terminals, Near and far terminals), SDMA, FDMA, TDMA, CDMA.

UNIT –III Mobile Network Layer:
Mobile IP (Goals, assumptions, entities and terminology, IP packet delivery, agent advertisement and discovery, registration, tunneling and encapsulation, optimizations), Dynamic Host Configuration Protocol (DHCP).

UNIT – IV Mobile Transport Layer:
Traditional TCP, Indirect TCP, Snooping TCP, Mobile TCP, Fast retransmit/fast recovery, Transmission /time-out freezing, Selective retransmission, Transaction oriented TCP.

UNIT – V Database Issues:
Hoarding techniques, caching invalidation mechanisms, client server computing with adaptation, power-aware and context-aware computing, transactional models, query processing, recovery, and quality of service issues.

UNIT – VI Data Dissemination:
Communications asymmetry, classification of new data delivery mechanisms, push-based mechanisms, pull-based mechanisms, hybrid mechanisms, selective tuning (indexing) techniques.
UNIT – VII Mobile Ad hoc Networks (MANETs):
Overview, Properties of a MANET, spectrum of MANET applications, routing and various routing algorithms, security in MANETs.

UNIT – VIII Protocols and Tools:
Wireless Application Protocol-WAP. (Introduction, protocol architecture, and treatment of protocols of all layers), Bluetooth (User scenarios, physical layer, MAC layer, networking, security, link management) and J2ME.

Text Books:


Reference Books:
UNIT-I
Introduction to Device Drivers: Role of Device Drivers, splitting the kernel, classes of Devices and modules, security issues, version numbering, building and running modules Kernel modules Vs. Applications, compiling & loading, kernel symbol table, preliminaries, Interaction and shutdown, module parameters, doing it in user space.

UNIT-II
Char Drivers: Design of scull, major and minor numbers, important data structures, char device registration, open and release, memory usage, read, write, playing with the new devices.

UNIT-III
Introduction to the Kernel – Important Data structures, Main Algorithms, Implementation of system calls. Debugging Techniques: Design support in the Kernel, debugging by printing, querying, watching, system faults..

UNIT-IV
Memory Management: Architecture Independent memory models, Virtual address space of a process, block device caching, Paging under Linux. Allocating memory – Kmalloc, lookaside caches, get free page and friends, vmalloc and friends, per – CPU variables, obtaining large Buffers.

UNIT-V
Concurrency and race Conditions: Pitfalls in scull, concurrency & its management, semaphores and mutexes, completions, spin locks, loading traps, alternatives to Locking.

UNIT-VI
Time, Delays, Deferred Work: Measuring time lapses, Knowing current time, delaying execution, kernel timers, tasklets, workqueues.

UNIT-VII
Interrupt handling: Preparing the parallel port, installing an Interrupt handler, implementing a handler, Top and bottom Halves, Interrupt Sharing, interrupt driven I/O. Communicating with H/W: I/O ports and I/O Memory, Using I/O ports, An I/O port example, using I/O memory.
UNIT-VIII
Data types in Kernel: Uses of structured C types, assigning an explicit size to data items, interface specific types, other portability issues, linked lists.

Text Books:

References:
Running Linux, 4th edition, Welsh et al, O’Reilly
MC 4.7 DATA MINING AND WAREHOUSING LAB

The objective of the lab exercises is to use data mining techniques to identify customer segments and understand their buying behavior and to use standard databases available to understand DM processes using WEKA (or any other DM tool)

1. Gain insight for running pre-defined decision trees and explore results using MS OLAP Analytics.

2. Using IBM OLAP Miner – Understand the use of data mining for evaluating the content of multidimensional cubes.

3. Using Teradata Warehouse Miner – Create mining models that are executed in SQL.

( BI Portal Lab: The objective of the lab exercises is to integrate pre-built reports into a portal application )

4. Publish cognos cubes to a business intelligence portal.

Metadata & ETL Lab: The objective of the lab exercises is to implement metadata import agents to pull metadata from leading business intelligence tools and populate a metadata repository. To understand ETL processes

5. Import metadata from specific business intelligence tools and populate a metadata repository.

6. Publish metadata stored in the repository.
7. **Load data from heterogenous sources including text files into a pre-defined warehouse schema.**

Case study

9. Design a data mart from scratch to store the credit history of customers of a bank. Use this credit profiling to process future loan applications.

10. Design and build a Data Warehouse using bottom up approach titled ‘Citizen Information System’. This should be able to serve the analytical needs of the various government departments and also provide a global integrated view.
III Year MCA  V Semester

MC 5.1 E- COMMERCE

UNIT-I
Electronic Commerce-Frame work, anatomy of E-Commerce applications, E-Commerce Consumer applications, E-Commerce organization applications.

UNIT-II
Consumer Oriented Electronic commerce - Mercantile Process models

UNIT-III
Electronic payment systems - Digital Token-Based, Smart Cards, Credit Cards, Risks in Electronic Payment systems.

UNIT-IV
Inter Organizational Commerce - EDI, EDI Implementation, Value added networks.

UNIT-V
Intra Organizational Commerce - work Flow, Automation Customization and internal Commerce, Supply chain Management.

UNIT-VI

UNIT-VII
Consumer Search and Resource Discovery - Information search and Retrieval, Commerce Catalogues, Information Filtering.

UNIT VIII
Multimedia - key multimedia concepts, Digital Video and electronic Commerce, Desktop video processing, Desktop video conferencing.
TEXT BOOKS:


REFERENCE BOOKS:

1. E-Commerce fundamentals and applications Hendry Chan, Raymond Lee, Tharam Dillon, Elizabeth Chang, John Wiley.
UNIT-I


UNIT-II

Fundamental concepts in video and digital audio: Types of video signals, analog video, digital video, digitization of sound, MIDI, quantization and transmission of audio.

UNIT-III

Action Script I: ActionScript Features, Object-Oriented ActionScript, Datatypes and Type Checking, Classes, Authoring an ActionScript Class

UNIT-IV

Action Script II: Inheritance, Authoring an ActionScript 2.0 Subclass, Interfaces, Packages, Exceptions

UNIT-V

Application Development: An OOP Application Frame work, Using Components with ActionScript MovieClip Subclasses.

UNIT VI

Multimedia data compression: Lossless compression algorithm: Run-Length Coding, Variable Length Coding, Dictionary Based Coding, Arithmetic Coding, Lossless Image Compression, Lossy compression algorithm: Quantization, Transform Coding, Wavelet-
Based Coding, Embedded Zerotree of Wavelet Coefficients Set Partitioning in Hierarchical Trees (SPIHT).

UNIT VII

Basic Video Compression Techniques: Introduction to video compression, video compression based on motion compensation, search for motion vectors, MPEG, Basic Audio Compression Techniques.

UNIT-VIII

Text Books:

1) Fundamentals of Multimedia by Ze-Nian Li and Mark S. Drew PHI/Pearson Education
2) Essentials ActionScript 2.0, Colin Moock, SPD O,REILLY.

Reference Books:

1) Digital Multimedia, Nigel chapman and jenny chapman, Wiley-Dreamtech
3) Multimedia and communications Technology, Steve Heath, Elsevier(Focal Press)
4) Multimedia Applications, Steinnetz, Nahrstedt, Springer.
5) Multimedia Basics by Weixel Thomson
6) Multimedia Technology and Applications, David Hilman, Galgotia
MC 5.3 OBJECT ORIENTED ANALYSIS AND DESIGN (Using UML)

UNIT-I
Introduction to UML: Importance of modeling, principles of modeling, object oriented modeling, conceptual model of the UML, Architecture, Software Development Life Cycle.

UNIT-II
Basic Structural Modeling: Classes, Relationships, common Mechanisms, and diagrams.
Advanced Structural Modeling: Advanced classes, advanced relationships, Interfaces, Types and Roles, Packages.

UNIT-III
Class & Object Diagrams: Terms, concepts, modeling techniques for Class & Object Diagrams.

UNIT- IV

UNIT-V
Basic Behavioral Modeling-II: Use cases, Use case Diagrams, Activity Diagrams.

UNIT-VI
Advanced Behavioral Modeling: Events and signals, state machines, processes and Threads, time and space, state chart diagrams.

UNIT-VII
Architectural Modeling: Component, Deployment, Component diagrams and Deployment diagrams.

UNIT-VIII
Case Study: The Unified Library application
TEXT BOOKS:

2. Hans-Erik Eriksson, Magnus Penker, Brian Lyons, David Fado: UML 2 Toolkit, WILEY-Dreamtech India Pvt. Ltd.

REFERENCE BOOKS:

MC 5.4 SOFTWARE PROJECT MANAGEMENT (Elective – III)

UNIT-I
Conventional Software Management: The waterfall model, conventional software Management performance.

UNIT-II
The old way and the new: The principles of conventional software Engineering, principles of modern software management, transitioning to an iterative process.

UNIT-III
Life cycle phases: Engineering and production stages, inception, Elaboration, construction, transition phases.
Artifacts of the process: The artifact sets, Management artifacts, Engineering artifacts, programmatic artifacts.

UNIT-IV
Model based software architectures: A Management perspective and technical perspective.
Work Flows of the process: Software process workflows, Iteration workflows,

UNIT-V
Checkpoints of the process: Major mile stones, Minor Milestones, Periodic status assessments.
Iterative Process Planning: work breakdown structures, planning guidelines, cost and schedule estimating, Iteration planning process, Pragmatic planning.

UNIT-VI
UNIT-VII

**Project Control and Process instrumentation:** The seven core Metrics, Management indicators, quality indicators, life cycle expectations, pragmatic Software Metrics, Metrics automation.

**Tailoring the Process:** Process discriminants.

UNIT-VIII

**Future Software Project Management:** modern Project Profiles, Next generation Software economics, modern process transitions.

**Case Study:** The command Center Processing and Display system- Replacement (CCPDS-R)

TEXT BOOKS:


REFERENCE BOOKS:

Software Project Management, Joel Henry, Pearson Education.
MC5.4.2 MIDDLE WARE TECHNOLOGIES

UNIT-I: 
**Introduction to client server computing:** Evolution of corporate computing models from centralized to distributed computing, client server models. Benefits of client server computing, pitfalls of client server programming.

UNIT-II: 
**CORBA with Java:** Review of Java concept like RMI, RMI API, JDBC. Client/Server CORBA-style, The object web: CORBA with Java.

UNIT III: 
**Introducing C# and the .NET Platform:** Understanding .NET Assemblies; Object – Oriented Programming with C#; Callback Interfaces, Delegates, and Events.

UNIT IV: 
**Building c# applications:** Type Reflection, Late Binding, and Attribute-Based Programming; Object Serialization and the .NET Remoting Layer; Data Access with ADO.NET; XML Web Services.

UNIT-V: 
**Core CORBA / Java:** Two types of Client/ Server invocations-static, dynamic. The static CORBA, first CORBA program, ORBlets with Applets, Dynamic CORBA-The portable count, the dynamic count multi count.

UNIT-VI: 
**Existential CORBA:** CORBA initialization protocol, CORBa activation services, CORBAIDL mapping CORBA java- to- IDL mapping, The introspective CORBA/Java object.

UNIT-VII: 
**Java Bean Component Model:** Events, properties, persistency, Intrespection of beans, CORBA Beans

UNIT-VIII: 
**EJBs and CORBA:** Object transaction monitors CORBA OTM’s, EJB and CORBA OTM’s, EJB container frame work, Session and Entity Beans, The EJB client/server development Process The EJB container protocol, support for transaction EJB packaging EJB design Guidelines.
**Text Books:**


3. C# and the .NET Platform Andrew Troelsen, Apress Wiley-dreamtech, India Pvt Ltd

**Reference: Books:**

- Distributed Computing, Principles and applications, M.L.Liu, Pearson Education
- Client/Server Computing D T Dewire, TMH.
- IBM Webspere Starter Kit Ron Ben Natan Ori Sasson, TMh, New Delhi
- Programming C#, Jesse Liberty, SPD-O’Reilly.
- C# Preciesely Peter Sestoft and Henrik I. Hansen, Prentice Hall of India
- Introduction to C# Using .NET Pearson Education
- C# How to program, Pearson Education
UNIT-I
Introduction: Definition, Objectives, Functional Overview, Relationship to DBMS, Digital libraries and Data Warehouses.

UNIT-II
Information Retrieval System Capabilities: Search, Browse, Miscellaneous

UNIT-III

UNIT-IV

UNIT-V
Automatic Indexing: Classes of automatic indexing, Statistical indexing, Natural language, Concept indexing, Hypertext linkages

UNIT-VI
Document and Term Clustering: Introduction, Thesaurus generation, Item clustering, Hierarchy of clusters.

UNIT-VII
User Search Techniques: Search statements and binding, Similarity measures and ranking, Relevance feedback, Selective dissemination of information search, Weighted searches of Boolean systems, Searching the Internet and hypertext. Information Visualization: Introduction, Cognition and perception, Information visualization technologies.

UNIT-VIII
TEXTBOOKS:


REFERENCE BOOKS:

Modern Information Retrieval By Yates Pearson Education.
III Year MCA  V Semester

MC 5.5.1 SOFTWARE TESTING METHODOLOGIES (ELECTIVE – IV)

UNIT-I:

**Introduction**:- Purpose of testing, Dichotomies, model for testing, consequences of bugs, taxonomy of bugs

UNIT-II:

**Flow graphs and Path testing**: Basics concepts of path testing, predicates, path predicates and achievable paths, path sensitizing, path instrumentation, application of path testing.

UNIT-III:

**Transaction Flow Testing**: transaction flows, transaction flow testing techniques. Dataflow testing:- Basics of dataflow testing, strategies in dataflow testing, application of dataflow testing.

UNIT-IV:

**Domain Testing**: domains and paths, Nice & ugly domains, domain testing, domains and interfaces testing, domain and interface testing, domains and testability.

UNIT-V:

**Paths, Path products and Regular expressions**: path products & path expression, reduction procedure, applications, regular expressions & flow anomaly detection.

UNIT-VI:

**Logic Based Testing**: overview, decision tables, path expressions, kv charts, specifications.

UNIT-VII:
State, State Graphs and Transition testing:- state graphs, good & bad state graphs, state testing, Testability tips.

UNIT-VIII:

Graph Matrices and Application:-Motivational overview, matrix of graph, relations, power of a matrix, node reduction algorithm, building tools. (Student should be given an exposure to a tool like JMeter or Win-runner).

TEXT BOOKS:


REFERENCE BOOKS:

1. The craft of software testing - Brian Marick, Pearson Education.
2. Software Testing Techniques – SPD(Oreille)
UNIT-I
**Introduction:** Distributed Data Processing, Distributed Database System, Promises of DDBSs, Problem areas.

**Overview of Relational DBMS:** Relational Database Concepts, Normalization, Integrity rules, Relational data languages.

UNIT-II
**Distributed DBMS Architecture:** Architectural Models for Distributed DBMS, DDMBS Architecture.

**Distributed Database Design:** Alternative Design Strategies, Distribution Design issues, Fragmentation, Allocation.

UNIT-III
**Query Processing and Decomposition:** Query Processing Objectives, Characterization of query processors, layers of query processing, query decomposition, Localization of distributed data.

UNIT-IV
**Distributed Query Optimization:** Query optimization, centralized query optimization, Distributed query optimization algorithms.

UNIT-V

UNIT-VI
**Distributed DBMS Reliability:** Reliability concepts and Measures, fault-tolerance in Distributed systems, failures in Distributed DBMS, local & Distributed Reliability Protocols, site failures and Network partitioning.

**Parallel Database Systems:** Database Series, Parallel Architecture, Parallel DBMS Techniques, Parallel exception problems, Parallel Execution for Hierarchical architecture.

UNIT-VII
**Distributed Object Database Management Systems:** Fundamental object concepts and Models, Object Distributed Design, Architectural Issues, Object Management, Distributed Object storage, Object query Processing.
UNIT VIII
Object Oriented Data Model: Inheritance, Object identity, persistent programming languages, persistence of objects, comparing OODBMS and ORDBMS

Text Books:
2. Stefano Ceri and Willipse Pelagatti: Distributed Databases, McGraw Hill.
3. Henry F Korth, A Silberchatz and Sudershan: Database System Concepts, MGH
4. Raghuramakrishnan and Johhanes Gehrke: Database Management Systems, MGH
MC 5.5.3 INFORMATION SECURITY (ELECTIVE – IV)

UNIT-I
Security Attacks ( Interruption, Interception, Modification and Fabrication), Security Services (Confidentiality, Authentication, Integrity, Non-repudiation, access Control and Availability) and Mechanisms, A model for Internetwork security, Internet Standards and RFCs, Buffer overflow & format string vulnerabilities, TCP session hijacking, ARP attacks, route table modification, UDP hijacking, and man-in-the-middle attacks.

UNIT-II
Conventional Encryption Principles, Conventional encryption algorithms, cipher block modes of operation, location of encryption devices, key distribution Approaches of Message Authentication, Secure Hash Functions and HMAC,

UNIT-III
Public key cryptography principles, public key cryptography algorithms, digital signatures, digital Certificates, Certificate Authority and key management Kerberos, X.509 Directory Authentication Service

UNIT-IV
Email privacy: Pretty Good Privacy (PGP) and S/MIME.

UNIT-V

UNIT-VI
Web Security Requirements, Secure Socket Layer (SSL) and Transport Layer Security (TLS), Secure Electronic Transaction (SET)

UNIT-VII
Basic concepts of SNMP, SNMPv1 Community facility and SNMPv3, Intruders, Viruses and related threats

UNIT-VIII
Firewall Design principles, Trusted Systems, Intrusion Detection Systems
TEXT BOOKS:


2. Hack Proofing your network by Ryan Russell, Dan Kaminsky, Rain Forest Puppy, Joe Grand, David Ahmad, Hal Flynn Ido Dubrawsky, Steve W.Manzuik and Ryan Permeh, wiley Dreamtech, 

REFERENCE BOOKS:

1. Fundamentals of Network Security by Eric Maiwald (Dreamtech press)