### III B.Tech I Semester

<table>
<thead>
<tr>
<th>Code</th>
<th>Subject</th>
<th>Periods per week</th>
<th>Credits</th>
<th>Scheme of Examination</th>
<th>Max. Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>L</td>
<td>T</td>
<td>P</td>
<td>Internal</td>
</tr>
<tr>
<td>10BT50501</td>
<td>Design and Analysis of Algorithms</td>
<td>4</td>
<td>1</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td>10BT61202</td>
<td>Computer Networks</td>
<td>4</td>
<td>-</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td>10BT50502</td>
<td>Microprocessors and Interfacing</td>
<td>4</td>
<td>1</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td>10BT50503</td>
<td>Database Management Systems</td>
<td>4</td>
<td>1</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td>10BT51201</td>
<td>Software Engineering</td>
<td>4</td>
<td>-</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td>10BT50504</td>
<td>Operating Systems</td>
<td>4</td>
<td>-</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td>10BT50511</td>
<td>Microprocessors and Interfacing Lab</td>
<td>-</td>
<td>-</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>10BT50512</td>
<td>Database Management systems Lab</td>
<td>-</td>
<td>-</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>10BT4HS02</td>
<td>Advanced English Communication Skills (Audit Course)</td>
<td>-</td>
<td>3</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

**TOTAL** 24 6 6 28 230 520 750

### III B.Tech II Semester

<table>
<thead>
<tr>
<th>Code</th>
<th>Subject</th>
<th>Periods per week</th>
<th>Credits</th>
<th>Scheme of Examination</th>
<th>Max. Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>L</td>
<td>T</td>
<td>P</td>
<td>Internal</td>
</tr>
<tr>
<td>10BT60501</td>
<td>Theory of computation</td>
<td>4</td>
<td>1</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td>10BT60502</td>
<td>UNIX Programming</td>
<td>4</td>
<td>-</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td>10BT60503</td>
<td>Data Warehousing and Data Mining</td>
<td>4</td>
<td>1</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td>10BT61201</td>
<td>Object Oriented Analysis and Design</td>
<td>4</td>
<td>-</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td>10BT60504</td>
<td>Distributed Computing</td>
<td>4</td>
<td>-</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td>10BT71204</td>
<td>Cryptography and Network Security</td>
<td>4</td>
<td>1</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td>10BT61211</td>
<td>Object Oriented Analysis and Design Lab</td>
<td>-</td>
<td>-</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>10BT60511</td>
<td>UNIX Programming Lab</td>
<td>-</td>
<td>-</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>10BT60512</td>
<td>Seminar</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>2</td>
</tr>
<tr>
<td>10BT60513</td>
<td>Knowledge Engineering Lab (Audit Course)</td>
<td>-</td>
<td>3</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

**TOTAL** 24 6 6 30 305 520 825
UNIT I: INTRODUCTION
Algorithm, Pseudo code for expressing algorithms, Performance Analysis-Space complexity, Time complexity, Asymptotic Notation-Big (o) notation, Omega notation, Theta notation and Little (o) notation, Recurrences, Probabilistic analysis.

UNIT II: DISJOINT SETS AND GRAPHS (Algorithm and Analysis)
Disjoint set operations, union and find algorithms, Graphs-Breadth First search and Traversal, Depth First Search and Traversal, spanning trees, connected components and biconnected components.

UNIT III: DIVIDE AND CONQUER
General method, Applications: Analysis of Binary search, Quick sort, Merge sort, Strassen’s matrix multiplication, Finding the Maxima and Minima.

UNIT IV : GREEDY METHOD
General method, Applications: Job sequencing with dead lines, 0/1 knapsack problem, Minimum cost spanning trees, Single source shortest path problem, Optimal storage on Tapes.

UNIT V: DYNAMIC PROGRAMMING
General method, Applications: Matrix chain multiplication, Optimal binary search trees, 0/1 knapsack problem, All pairs shortest path problem, Travelling sales person problem, Reliability design, String Editing.

UNIT VI: BACKTRACKING
General method, applications: n-queen problem, sum of subsets problem, graph colouring, 0/1 knapsack problem, Hamiltonian cycles.
UNIT VII: BRANCH AND BOUND
General method, applications - Travelling sales person problem, 0/1 knapsack problem- LC Branch and Bound solution, FIFO Branch and Bound solution.

UNIT VIII: NP-HARD AND NP-COMPLETE PROBLEMS
Basic concepts, non-deterministic algorithms, NP - Hard and NP Complete classes, Cook’s theorem, NP-hard scheduling Problems.

TEXT BOOK:

REFERENCE BOOKS:
UNIT I: INTRODUCTION

UNIT II: THE PHYSICAL LAYER
Theoretical Basis for communication, Guided Transmission media, Wireless Transmission, The public switched telephone Networks, Mobile telephone system.

UNIT III: THE DATA LINK LAYER

UNIT IV: THE MEDIUM ACCESS SUBLAYER
Channel Allocations problem, Multiple Access protocols: ALOHA, CSMA, CSMA/CD protocols, Collision free protocols, Limited contention protocols, Ethernet, DLL Switching.

UNIT V: THE NETWORK LAYER

UNIT VI: THE TRANSPORT LAYER
Transport Service, Elements of transport protocol, Internet Transport layer protocols: UDP and TCP.
UNIT VII: THE APPLICATION LAYER

UNIT VIII: IEEE STANDARDS AND NETWORK SECURITY
Introduction to IEEE standards, Wi-Fi: 802.11b, Bluetooth: 802.15, 3G: 802.16, 4G: 802.16m, Wi-Max: 802.16a.

TEXT BOOK:

REFERENCE BOOKS:
UNIT-I: INTRODUCTION

UNIT-II: ASSEMBLY LANGUAGE PROGRAMMING
Assembly language programs involving logical, branch and call instructions, sorting, evaluation of arithmetic expressions, string manipulation.

UNIT-III: ARCHITECTURE OF 8086 & INTERFACING
Pin configuration of 8086-Minimum mode and maximum mode of operation, Timing diagram. Memory interfacing to 8086 (static RAM and EPROM), Need of Direct Memory Access (DMA), DMA data transfer method, Interfacing with 8237/8257.

UNIT-IV: PROGRAMMABLE INTERFACING DEVICES
8255 PPI-various modes of operation and interfacing to 8086, Interfacing keyboard, displays, 8279, stepper motor and actuators, D/A and A/D converter interfacing.

UNIT-V: INTERRUPTS AND PROGRAMMABLE INTERRUPT CONTROLLERS
Interrupt structure of 8086, Interrupt Vector table. Interrupt service routines, Introduction to DOS and BIOS interrupts, 8259 PIC architecture and interfacing cascading of interrupt controller and its importance, Programming with 8259.

UNIT-VI: SERIAL DATA TRANSFER SCHEMES
Asynchronous and synchronous data transfer schemes, 8251 USART architecture and interfacing, TTL to RS232C and RS232C to TTL conversion, Sample programs for serial data transfer. Introduction to high-speed serial communications standards, USB.
UNIT-VII: ADVANCED MICROPROCESSORS
Introduction to 80286, Salient Features of 80386, Real and Protected Mode, Segmentation and Paging, Salient Features of Pentium, Branch Prediction, Overview of RISC Processors.

UNIT-VIII: 8051 MICROCONTROLLER AND ITS PROGRAMMING

TEXT BOOKS:

REFERENCE BOOKS:
1. Douglas V.Hall, Microprocessors Interfacing, 2 ed, 2007, TMH.
3. Liu and GA Gibson, Micro computer system 8066/8088 family Architecture, programming and Design, 2 ed, PHI.
III B.Tech. I Semester

10BT50503: DATABASE MANAGEMENT SYSTEMS

UNIT I: INTRODUCTION

UNIT II: DATABASE DESIGN
Introduction to Database Design and E-R Diagrams, Entities, Attributes and Entity Sets, Relationships and Relationship Sets, Additional Features of the E-R Model, Conceptual Design with the E-R Model, Conceptual Design for Large Enterprises.

UNIT III: THE RELATIONAL MODEL

UNIT IV: SQL: QUERIES, CONSTRAINTS, TRIGGERS
Overview, The form of a Basic SQL Query, Union, Intersect and Except operators, Nested Queries, Aggregate Operators, Null values, Complex Integrity Constraints in SQL, Triggers and Active Databases, Designing Active Databases.(Chapter 5;Sections 5.1-5.9 including subtopics from Text book-1)

UNIT V: SCHEMA REFINEMENT AND NORMAL FORMS
Introduction to Schema Refinement, Functional Dependencies, Reasoning about FDs, Normal Forms – 1NF, 2NF, 3NF, BCNF, Properties of Decompositions, Normalization, Schema Refinement in Database Design, Other Kinds of Dependencies – 4NF, 5NF, DKNF, Case Studies.
UNIT VI: TRANSACTIONS MANAGEMENT

UNIT VII: CONCURRENCY CONTROL AND RECOVERY SYSTEM.
Concurrency Control: Lock Based protocols, Time-Stamp Based Protocols, Validation based Protocols, Multiple Granularity, and Deadlock Handling.

UNIT VIII: OVERVIEW OF STORAGE AND INDEXING
Data on External Storage, File Organizations and Indexing, Index Data Structures, Comparison of File Organizations, Indexes and Performance Tuning.

TEXT BOOK:

REFERENCE BOOKS:
III B.Tech. I Semester

10BT51201: SOFTWARE ENGINEERING

UNIT-I: INTRODUCTION TO SOFTWARE ENGINEERING

The evolving role of software, Changing Nature of Software, Software myths.


UNIT-II: PROCESS MODELS


Software Requirements: Functional and Non-functional requirements, User requirements, System requirements, Interface specification, the software requirements document.

UNIT-III: REQUIREMENTS ENGINEERING PROCESSES

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.


UNIT-V: USER INTERFACE DESIGN AND RE-ENGINEERING

Performing User Interface Design: Golden rules, User interface analysis and design, interface analysis, interface design steps, Design evaluation.

Re-Engineering: Reverse Engineering, Restructuring, Forward Engineering.
UNIT-VI: SOFTWARE TESTING

A strategic approach to software testing, test strategies for conventional software, Black-Box and White-Box testing, System testing, the art of Debugging.

UNIT-VII: SOFTWARE METRICS

Product Metrics: Software Quality, Metrics for Analysis Model, Metrics for Design Model, Metrics for source code, Metrics for testing, Metrics for maintenance.

Size Oriented Metrics, Function-Oriented Metrics, Reconciling LOC and FP Metrics, Object-Oriented Metrics, Use-Case Oriented Metrics, Web Engineering Project Metrics, Metrics for Software Quality.

UNIT-VIII: RISK AND QUALITY MANAGEMENT


TEXT BOOKS:


REFERENCE BOOKS:


UNIT I: OPERATING SYSTEMS OVERVIEW
Introduction, Operating system operations, Process management, Memory management, Storage management, Protection and Security, Distributed Systems, Special purpose systems. Operating systems structures: Operating system services and Systems calls, System programs, Operating system structure, Operating systems generations.

UNIT II: PROCESS MANAGEMENT

UNIT III: CONCURRENCY AND SYNCHRONIZATION
Process synchronization, Critical-section problem, Peterson’s Solution, Synchronization Hardware, semaphores, Classic problems of synchronization, Readers and Writers problem, Dining-philosophers problem, Monitors, Synchronization examples(Solaris), atomic transactions. Comparison of UNIX and Windows.

UNIT IV: DEADLOCKS
System model, deadlock characterization, deadlock prevention, detection and avoidance, recovery from deadlock- bankers algorithm.

UNIT V: MEMORY MANAGEMENT
Swapping, contiguous memory allocation, paging, structure of the page table, segmentation, virtual memory, demand paging, page-replacement algorithms, Allocation of frames, Thrashing, case study-UNIX.

UNIT VI: FILE SYSTEM
Concept of a file, Access Methods, Directory structure, File system mounting, File sharing, protection.
File System implementation: File system structure, file system implementation, directory implementation, allocation methods, free-space management, efficiency and performance, comparison of UNIX and Windows.

UNIT VII: I/O SYSTEM
Mass-storage structure: Overview of Mass-storage structure, Disk structure, disk attachment, disk scheduling algorithms, swap-space management, stable-storage implementation, Tertiary storage structure.
I/O: Hardware, application I/O interface, kernel I/O subsystem, Transforming I/O requests to Hardware operations, STREAMS, performance.

UNIT VIII: PROTECTION AND SECURITY

TEXT BOOK:

REFERENCE BOOKS:
I. Microprocessor 8086:
1. Introduction to MDS.
2. Arithmetic operation – Multi byte Addition and Subtraction, Multiplication and Division – Signed and unsigned Arithmetic operation, ASCII – arithmetic operation.
3. Logic operations – Shift and rotate – Converting packed BCD to unpacked BCD, BCD to ASCII conversion.
4. By using string operation and Instruction prefix: Move Block, Reverse string, Sorting,Inserting, Deleting, Length of the string, String comparison.
5. DOS/BIOS programming: Reading keyboard (Buffered with and without echo) – Display characters, Strings.

II. Interfacing:
1. 8259 – Interrupt Controller: Generate an interrupt using 8259.
2. 8279 – Keyboard Display: Write a small program to display a string of characters.
3. 8255 – PPI: Interfacing DAC, Stepper Motor, ADC.
4. 8251 – USART: Write a program in ALP to establish Communication between two processors.

III. Microcontroller 8051
1. Reading and Writing on a parallel port.
2. Timer in different modes.
3. Serial communication implementation.
DESCRIPTION OF SALES DATABASE

ABC is a company operating in the country with a chain of shopping centers in various cities. Everyday large numbers of items are sold in different shopping centers. The Sales database comprises of various tables like CUST, PROD, SALES_DETAIL, STATE_NAME with the following schemas.

CUST TABLE

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>CID</td>
<td>VARCHAR2(6)</td>
<td>PRIMARY KEY</td>
</tr>
<tr>
<td>CNAME</td>
<td>VARCHAR2(10)</td>
<td></td>
</tr>
<tr>
<td>CCITY</td>
<td>VARCHAR2(8)</td>
<td></td>
</tr>
</tbody>
</table>

PROD TABLE

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>PID</td>
<td>VARCHAR2(6)</td>
<td>PRIMARY KEY</td>
</tr>
<tr>
<td>PNAME</td>
<td>VARCHAR2(6)</td>
<td></td>
</tr>
<tr>
<td>PCOST</td>
<td>NUMBER(4,2)</td>
<td></td>
</tr>
<tr>
<td>PPROFIT</td>
<td>NUMBER(3)</td>
<td></td>
</tr>
</tbody>
</table>

SALES_DETAIL

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>CID</td>
<td>VARCHAR2(6)</td>
<td>COMPOSITE PRIMARY KEY</td>
</tr>
<tr>
<td>PID</td>
<td>VARCHAR2(6)</td>
<td>COMPOSITE PRIMARY KEY</td>
</tr>
<tr>
<td>SALE</td>
<td>NUMBER(3)</td>
<td>COMPOSITE PRIMARY KEY</td>
</tr>
<tr>
<td>SALEDT</td>
<td>DATE</td>
<td>COMPOSITE PRIMARY KEY</td>
</tr>
</tbody>
</table>

STATE_NAME

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>CCITY</td>
<td>VARCHAR2(8)</td>
<td>PRIMARY KEY</td>
</tr>
<tr>
<td>STATE</td>
<td>VARCHAR2(15)</td>
<td></td>
</tr>
</tbody>
</table>
1. ER MODEL
Draw an ER Model indicating many to many relationship between CUST vs PROD. Show the Cardinality Ratio between PROD and SALES_DETAIL is one-to-many because one product can be sold multiple times. Similarly show the Cardinality Ratio between CUST and SALES_DETAIL is one-to-many because one customer can purchase many products. Indicate CID# and PID# are unique in CUST and PROD entity respectively, where as CID and PID in SALE_DETAIL entity may occur many times.
Represent the ER Model in Tabular Form.

2. NORMALIZATION
In the above relations the following Functional Dependencies exist:
CID \( \rightarrow \) CNAME, CCITY, STATE
PID \( \rightarrow \) PNAME, PCOST, PPROFIT
CID, PID, SALEDT \( \rightarrow \) SALE

<table>
<thead>
<tr>
<th>CID#</th>
<th>CNAME</th>
<th>CCITY</th>
<th>STATE</th>
<th>PID#</th>
<th>PNAME</th>
<th>PCOST</th>
<th>PROFIT</th>
<th>SALE</th>
<th>SALEDT#</th>
</tr>
</thead>
<tbody>
<tr>
<td>C1</td>
<td>RAVI</td>
<td>HYD</td>
<td>AP</td>
<td>P1</td>
<td>CD</td>
<td>10</td>
<td>5</td>
<td></td>
<td>14-JUL-10</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>P3</td>
<td>DVD</td>
<td>20</td>
<td>10</td>
<td>2</td>
<td>14-JUL-10</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>P3</td>
<td>DVD</td>
<td>20</td>
<td>10</td>
<td>3</td>
<td>20-AUG-09</td>
</tr>
</tbody>
</table>

Normalize the above table into 1NF, 2NF and 3NF. And handle Insert, Delete and Update anamolies.

3. DATA RETRIEVAL
a) Write a query to display all columns of CUST table.
b) Write a query to display pname of all records. Sort all records by pname. (use order by clause)
c) Write a query to display cname and ccity of all records. Sort by ccity in descending order.
d) Write a query to display cname, ccity who lives in mysore.
e) Write a query to display cname, pname, sale, saledt for all customers.
f) Write a query to display cname who have purchased Pen.
g) Write a query to display saledt and total sale on the date labeled as sale of all items sold after 01-sep-2010.
h) Write a query to display sold on the date labeled as sale of all items other than DVD.
i) Write a query to display cname and ccity of all customers who live in Kolkata or Chennai.

4. USE OF DISTINCT, BETWEEN, IN CLAUSE, LIKE OPERATOR, DUAL
   a) Write a query to display the pname and pcost of all the customers where pcost lies between 5 and 25.
   b) Find the product ids in sale_detail table (eliminating duplicates).
   c) Write a query to display distinct customer id where product id is p3 or sale date is '18-mar-2011'.
   d) Write a query to display cname, pid and soldt of those customers whose cid is in c1 or c2 or c4 or c5.
   e) Write a query to display cname, pid, soldt of those customers whose pid is p3 or sale date is '20-dec-2009'.
   f) Write a query to display system date.
   g) Write a query to display all records of prod table in which first and third character of pname is any character and second character is 'E'.
   h) Write a query to display all cname which includes two 'A' in the name.

5. CONSTRAINTS
   a) Implement table level and Column level constraints like NOT NULL, UNIQUE, PRIMARY KEY, FOREIGN KEY, CHECK.

6. SINGLE ROW FUNCTIONS: DATE FUNCTION
   a) Write a query to display the system date by rounding it to next month.
   b) Write a query to display the system date by rounding it to next year.
   c) Write a query to display the last date of the system date.
   d) Write a query to display the next date of system date which is Friday.
   e) Write a query to display sale date and date after 02 months from sale date.
f) Write a query to display system date, sale date and months between two dates.

g) Write a query to display the greatest date between sale date and system date, name it as BIG, also display sale date and SYSDATE.

h) Write a query to display the least date between sale date and system date name it as SMALL, also display sale date and SYSDATE.

7. SINGLE ROW FUNCTIONS: NUMERIC AND CHARACTER FUNCTION

a) Write a query to display the product name along with the rounded value of product cost for product name is “Pencil”.

b) Write a query to display product cost along with MOD value if divided by 5.

c) Write a query to display cname in uppercase, lowercase, titlecase from cust table where customer name is “rohan”.

d) Write a query to display all concatenated value of cname, ccity by converting cname into titlecase and ccity into uppercase.

e) Write a query to display the first 3 characters of cname.

f) Write a query to display the position of ‘M’ in the cname of the customer whose name is “SAMHITA”.

g) Write a query to display the length of all customer names.

h) PAD # character in left of product cost to a total width of 5 character position.

8. GROUP FUNCTIONS AND SET FUNCTIONS

a) Write a query to display the total count of customer.

b) Write a query to display the minimum cost of product.

c) Write a query to display average value of product cost rounded to 2nd decimal places.

d) Write a query to display product name with total sale detail in descending order.

e) Write a query to display product name, sale date and total amount collected for the product.

f) Write a query to display sale date and total sale date wise which was sold after “14-jul-08”.
g) Write a query to display the customer name who belongs to those places whose name is having I or P.

h) Write a query to display customer name who belongs to a city whose name contains characters ‘C’ and whose name contains character ‘A’.

i) Write a query to display the customer name who does not belong to PUNE.

9. PL/SQL

a) Write a PL/SQL program to find largest number among three.
   (Hint: Use Conditional Statement)

b) Write a PL/SQL program to display the sum of numbers from 1 to N using for loop, loop...end and while...loop.

10. SQL CURSOR

a) Write a PL/SQL program to display the costliest and cheapest product in PROD table.

b) Write a PL/SQL program which will accept PID and display PID and its total sale value i.e. sum.

11. FUNCTIONS

a) Write a function that accepts two numbers A and B and performs the following operations.
   i. Addition
   ii. Subtraction
   iii. Multiplication
   iv. Division

b) Write a function that accepts to find the maximum PCOST in PROD table.

12. PROCEDURES

a) Write a procedure that accepts two numbers A and B, add them and print.

b) Write procedures to demonstrate IN, IN OUT and OUT parameter.
13. TRIGGER
   a) Develop a PL/SQL program using BEFORE and AFTER triggers.

14. CURSOR
   a) Declare a cursor that defines a result set. Open the cursor to establish the result set. Fetch the data into local variables as needed from the cursor, one row at a time. Close the cursor when done.
UNIT I: VOCABULARY BUILDING:
Synonyms and Antonyms, Word roots, One-word substitutes, Prefixes and Suffixes, Study of word origin, Analogy, Idioms and Phrases.
Functional English: starting conversation, responding appropriately and relevantly, using the right body language, role play in different situations.

UNIT II: READING COMPREHENSION
Reading for facts, Guessing meanings from context, Scanning, Skimming, Inferring meaning and Critical reading.

UNIT III: ACADEMIC ESSAY WRITING
Accuracy, Brevity, Clarity, Brainstorm, List your ideas, Sub-headings, Revising Content and Organisation.

Unit IV: TECHNICAL REPORT WRITING
Types of formats and styles, Subject-matter, Subject-organization, Clarity, Coherence and Style, Planning, Data-collection, Tools, Analysis.

Unit V: CAREER SKILLS
Career direction, Exploring your talents, Personality inventories, Write a "Who I Am" statement, Thinking further, Perform career research, How do I get hired, Creating job satisfaction, Identify your satisfaction triggers, Positive attitude, Maintain a balanced lifestyle, Analyze your job in terms of your interests, Set goals to bring your interests and responsibilities in line, Personal SWOT analysis, Making the most of your talents and opportunities, Shaping your job to fit you better, Future proof your career, Managing your emotions at work, Get the recognition you deserve.
UNIT VI: RESUME WRITING
Structure and Presentation, Planning, Defining the career objective, Projecting ones strengths and skill-sets, Summary, Formats and Styles, Cover letter.

UNIT VII : GROUP DISCUSSION
Dynamics of group discussion, Intervention, Summarizing, Modulation of voice, Fluency and Coherence, Participation, Relevance, Assertiveness, Eye contact and Body language.

Unit VIII: INTERVIEW SKILLS
Concept and Process, Pre-interview planning, Opening strategies, Answering strategies, Interview through Tele and Video-conferencing.

REFERENCE BOOKS:

SUGGESTED SOFTWARE:
1. TOEFL, GRE and IELTS (Kaplan, Aarco and Barrons, Cliffs)
2. Softwares from 'train2success.com'
5. Study Skills Success, (Essay, Vocabulary strategies, IELTS), Young India Films.
6. Vocabulary Builder, Young India Films.
7. E-correspondence, Young India Films.
8. Group Discussions, (Ease - 2), Young India Films.
UNIT–I: INTRODUCTION TO THEORY OF AUTOMATA
Strings, Alphabets, Language, Operations on sets, Definition of an automaton, Description of a Finite Automaton (FA), Transition systems, Properties of transition functions, Acceptability of a string by a finite automaton.

UNIT–II: FINITE AUTOMATA
Deterministic finite automata(DFA), Nondeterministic finite automata(NFA), The language of a DFA, The Language of an NFA, NFA with ε-transitions, Equivalence between NFA with and without ε-transitions, NFA to DFA conversion, Equivalence between two finite state machines, Finite automata with output-Mealy and Moore machines, Minimization of finite automata.

UNIT–III: REGULAR EXPRESSIONS
Regular expressions, Regular sets, Identity rules, Constructing finite automata for a given regular expressions, Conversion of finite automata to regular expressions, Pumping lemma for regular sets, Applications of pumping lemma, Closure properties of regular sets.

UNIT–IV: FORMAL LANGUAGES
Basic definitions and examples, Chomsky classification of languages, Languages and their relation, Languages and automata, Regular grammars- Right linear and Left linear grammars, Equivalence between regular linear grammar and FA.
Context Free Grammars: Definition of context free grammars(CFG), Leftmost and rightmost derivations, The language of a grammar, Sentential forms, Constructing parse trees, The yield of a parse tree, Ambiguous grammars, Removing ambiguity from grammars.
UNIT-V: CONTEXT FREE LANGUAGES
Simplification of CFG, Eliminating useless symbols, Elimination of NULL productions, Elimination of unit productions, Chomsky Normal Form (CNF), Greibach Normal Form (GNF), Pumping lemma for context free languages (CFL).

UNIT-VI: PUSHDOWN AUTOMATA
Definition of pushdown automaton (PDA), The Languages of a PDA, Equivalence of PDA's and CFG's, Deterministic pushdown automaton.

UNIT-VII: TURING MACHINES AND LINEAR BOUNDED AUTOMATA
Turing Machine model, Representation of Turing Machines (TM), Languages acceptability by Turing Machines, Design of Turing Machines, Computable functions, Recursively enumerable languages, Church's hypothesis, Counter machine, Types of Turing Machines, The model of linear bounded automaton (LBA), Turing Machines and type 0 grammar, Linear bounded automata and Languages.

UNIT-VIII: COMPUTABILITY THEORY
LR(k) grammar, Universal Turing Machines, Undecidable problems about Turing Machines, Post's Correspondence Problem, The Classes P and NP, An NP-Complete and NP-Hard Problems.

TEXT BOOK:

REFERENCE BOOKS:
UNIT-I: INTRODUCTION TO UNIX AND UNIX UTILITIES
A Brief history of Unix, Architecture of Unix, Features of Unix, Introduction to vi editor. General Purpose Utilities, File Handling Utilities, Security by File Permissions, Process Utilities, Disk Utilities, Networking Commands, detailed commands to be covered are passwd, tty, script, clear, date, cal, cp, mv, ln, rm, unlink, mkdir, rmdir, du, df, mount, umount, find, unmask, ulimit, ps, who, w, finger, arp, ftp, telnet, rlogin.

UNIT-II: TEXT PROCESSING AND BACKUP UTILITIES
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-III: WORKING WITH THE BOURNE AGAIN SHELL (BASH)
Shell, Shell Responsibilities, Types of Shell, Pipes and I/O Redirection, Shell as a Programming Language, Shell Syntax: Variables, Conditions, Control Structures, Commands, Command Execution, Here Documents, and Debugging Scripts.

UNIT-IV: UNIX FILE STRUCTURE
UNIT-V: PROCESS AND SIGNALS

UNIT-VI: DATA MANAGEMENT AND FILE LOCKING
Data Management: Managing Memory: malloc, free, realloc, calloc, File Locking: Creating Lock Files, Locking Regions, Use of Read and Write with Locking, Competing Locks, Other Lock Commands-Advisory Locking,Mandatory Locking;Deadlocks.

UNIT-VII: INTER-PROCESS COMMUNICATION

UNIT-VIII: INTRODUCTION TO SOCKETS
Socket, Socket Connections - Socket Attributes, Socket Addresses, socket, connect, bind, listen, accept, Socket Communications.

TEXT BOOK:

REFERENCE BOOKS:
UNIT-I: DATA WAREHOUSE AND OLAP TECHNOLOGY
Data Warehouses – Definitions – Multidimensional Data Model – Data Warehouse Architecture.(Chapter 3;Sections 3.1-3.3 including sub topics of the Text book)

UNIT-II: INTRODUCTION TO DATA MINING
Definition of Data Mining – Kinds of Data – Data Mining Functionalities– Classification of Data Mining Systems – Primitives – Major Issues in Data Mining.

UNIT-III: DATA PREPROCESSING
Descriptive Data Summarization- Data Cleaning – Data Integration and Transformation – Data Reduction – Data Discretization and Concept Hierarchy Generation.

UNIT-IV: MINING FREQUENT PATTERNS AND ASSOCIATIONS
Basic Concepts – Efficient and Scalable Frequent Itemset Mining Methods – Association Rule Mining.(Chapter 5;Sections 5.1-5.5 including subtopics of Text book)

UNIT-V: CLASSIFICATION
Decision Tree Induction, Bayesian Classification – Rule Based Classification, Prediction – Accuracy and Error Measures.

UNIT-VI: CLUSTER ANALYSIS
Cluster Analysis – Categories of Clustering Methods – Partitioning Methods – Hierarchical Methods – Density based Methods – Grid based methods – Model Based Clustering methods – Clustering High Dimensional Data – CLIQUE.

UNIT-VII: MINING STREAM, TIME SERIES AND SEQUENCE DATA
Mining data streams, Mining Time Series Data, Mining Sequence Patterns in Biological Data.
UNIT-VIII: MINING OBJECT, SPATIAL, MULTIMEDIA, TEXT AND WEB

Multi Dimensional Analysis on Complex Object data types – Descriptive Mining on Complex Objects – Spatial Data Mining – Multimedia Data Mining – Text Mining – Web Mining.

TEXT BOOK:

REFERENCE BOOKS:
III B.Tech. II Semester

10BT61201: OBJECT ORIENTED ANALYSIS AND DESIGN

UNIT-I: INTRODUCTION TO UML
Introduction to object oriented concepts like inheritance, polymorphism, information hiding, Importance of modeling, principles of modeling, object oriented modeling, An overview of UML, conceptual model of the UML, Architecture, Software Development Life Cycle.

UNIT-II: BASIC STRUCTURAL MODELING
Classes-Terms and concepts, Common modeling techniques, Relationships-modeling simple dependencies, single Inheritance and structural relationships, common Mechanisms, and diagrams. Advanced Structural Modeling: Advanced classes, advanced relationships, Interfaces, Types and Roles, Packages, Instances.

UNIT-III: CLASS AND OBJECT DIAGRAMS
Terms, concepts, modeling techniques for Class Diagram-modeling Simple collaboration, Logical database Schema Forward and Reverse Engineering, Object Diagrams-Modeling object structures, Forward and reverse engineering.

UNIT-IV: BASIC BEHAVIORAL MODELING-I
Interactions-Terms and concepts, modeling a flow of control, Interaction diagrams-terms and concepts, modeling flows of control by time ordering and control by organization, Forward and reverse Engineering.

UNIT-V: BASIC BEHAVIORAL MODELING-II
Use cases-terms and concepts, modeling the behavior of the element, Usecase Diagrams-Terms and concepts, modeling the context of a system and requirement of a system, Forward and reverse Engineering, Activity Diagrams-Terms and concepts, modeling a workflow and an operation, Forward and reverse Engineering.
UNIT-VI: ADVANCED BEHAVIORAL MODELING
Events and signals-modeling a family of signals and exceptions, state machines-modeling the lifetime of an object, state machines, processes and Threads-modeling multiple flows of control and interprocess communication, time and space-modeling timing constraints, distribution of objects and objects that migrate, state chart diagrams-modeling reactive objects and Forward and reverse Engineering.

UNIT-VII: ARCHITECTURAL MODELING
Component-Terms and concepts,modeling executables and Libraries, modeling tables, .file, and documents, modeling an API, modeling source code, Deployment-modeling processors and devices, modeling the distribution of components, Component diagrams-modeling source code, executable release, physical database, Adaptable Systems, Forward and reverse Engineering and Deployment diagrams-modeling an embedded systems, Client/server System, Fully distributed systems, Forward and reverse Engineering.

UNIT-VIII: CASE STUDIES
Model all the views of: Automation of a Library, Point of Sales System.

TEXT BOOK:

REFERENCE BOOKS:
III B.Tech. II Semester

10BT60504: DISTRIBUTED COMPUTING

UNIT-I: INTRODUCTION
Forms of computing-Strengths and weaknesses of distributed computing-OS overview-Network overview-Software Eng. overview.

UNIT-II: INTERPROCESS COMMUNICATION
IPC program interface-Event synchronization-Timeouts and threading- Deadlock and timeouts-Data representation- Data encoding-Text based protocols-Request response protocols-Event and sequence diagram-Connection vs connectionless IPC.

UNIT-III: DISTRIBUTED COMPUTING PARADIGMS
Message passing, client server, peer to peer, message system, remote procedure, call model, distributed objects, object space, mobile agent, network services, collaborative application - Abstraction, Tradeoffs: abstraction vs overhead, scalability, cross-platform.

UNIT-IV: SOCKET API
Socket metaphor, diagram socket API stream mode socket API, sockets with non-blocking I/O, secure socket API, Client server paradigm, Issues, service session, protocol for a service, Inter-process communications & event synchronization, data representation, Software engineering for a network service, software architecture, IPS Mechanism, Daytime client server, Connection oriented and connectionless servers, Echo client server, Iterative server and concurrent server, Stateful servers - global state information, session state information.

UNIT-V: GROUP COMMUNICATION
Unicasting, Multicasting, Multicast API, Connection oriented and connectionless Reliable, Unreliable multicast, Java Basic Multicast API-IP Multicast addresses, Joining/sending multicast group.
Distributed Objects—message passing vs distributed object, distributed object architecture, distributed object systems, remote procedure calls, Java RMI architecture, client side, server side, object registry, API for Java RMI, Remote interface, server side software, client side software, RMI vs socket API.

UNIT-VI: REMOTE METHOD INVOCATION (RMI)
Internet applications—HTML, XML, HTTP, Client request, Server response Content type and MIME, HTTP: connection oriented, stateless protocol, Dynamically generated web contents, Common gateway interface, Web form, Query string processing Encoding and decoding query strings, Environment variables in CGI, Web session and session state data, hidden form feeds for transferring session state data, cookies for transferring session state data, HTTP header lines.

UNIT-VII: DATA PRIVACY AND SECURITY CONCERNS

UNIT-VIII: ADVANCED DISTRIBUTED COMPUTING PARADIGMS:
Message Queue system paradigm - Point to point, Publish/Subscribe, Mobile Agents - Basic architecture, Advantages, Mobile agent framework systems, Network services, Object spaces.

TEXT BOOK:
REFERENCE BOOKS:


UNIT - I: INTRODUCTION

UNIT - II: ENCRYPTION PRINCIPLES
Conventional encryption algorithms: Feistal structure, DES algorithm, S-Boxes, Triple DES, Advanced Data Encryption Standard (AES), Cipher block modes of operation, location of encryption devices, Key Distribution Approaches of Message Authentication, Secure Hash Functions and HMAC.

UNIT - III: CRYPTOGRAPHY AND APPLICATIONS

UNIT - IV: ELECTRONIC MAIL SECURITY
Email privacy: PGP operations, Radix-64 Conversion, Key Management for PGP, PGP Trust Model, Multipurpose Internet Mail Extension (MIME), Secure/MIME(S/MIME).

UNIT - V: IP SECURITY ARCHITECTURE AND SERVICES
UNIT – VI: WEB SECURITY
Web Security Considerations, Secure Socket Layer (SSL) and Transport Layer Security (TLS), Secure Electronic Transaction (SET).

UNIT - VII: NETWORK MANAGEMENT SECURITY

UNIT - VIII: FIREWALLS

TEXT BOOKS:

REFERENCE BOOKS:
III B.Tech. II Semester
10BT61211: OBJECT ORIENTED ANALYSIS AND DESIGN LAB

L T P C
- - 3 2

Case studies given below should be Modeled using Rational Rose tool in different views i.e Use case view, logical view, component view, Deployment view.

CASE STUDY 1: LIBRARY INFORMATION SYSTEM
Problem Statement:
A library lends books and magazines to members, who are registered in the system. Also it handles the purchase of new titles for the library. Popular titles are bought in multiple copies. A member can reserve a book or magazine that is not currently available in the library, so that when it is returned by the library that person is notified. The library can easily create, update and delete information about the titles, members, loans and reservations in the systems.

CASE STUDY 2: A POINT OF SALE (POS) SYSTEM
Problem Statement:
A POS System is a computerized application used to record sales and handle payments; it is typically used in a retail store. It includes hardware components such as a computer and bar code scanner, and software to run the system. It interfaces to various service applications, such as a third-party tax calculator and inventory control. These systems must be relatively fault tolerant; that is, even if remote services and temporarily unavailable they must still be of capturing sales and handling at least cash payments. A POS system must support multiple and varied client – side terminals and interfaces such as browser, PDA’s, touch – screens.

CASE STUDY 3: AUTOMATED TELLER MACHINE (ATM)
Problem Statement:
Software is designed for supporting a computerized ATM banking network. All the process involved in the bank is
computerized these days. All the accounts maintained in the bank and also the transactions effected, including ATM transactions are to be processed by the computers in the bank. An ATM accepts a relevant cash card, interacts with user, communicates with the central system to carry out the transaction, dispenses cash, and prints receipts. The system to be designed and implemented must include appropriate record keeping and security provisions. The system must handle concurrent access to the same account.

**CASE STUDY 4: ONLINE TICKET RESERVATION FOR RAILWAYS**

**Problem Statement:**

Computer play an integral part of the day in today’s life. It makes the entire job easier and faster, every job is computerized so as the ticket reservation we can book over the online ticket reservation system. During the booking of the ticket reservation passenger has to select origin, date of journey, destination, class of train etc. The reservation counter keeps track of passenger’s information. Thus the system will have all the details about the trains and facilities provided by them. There are various trains with the different level of convenience for the passengers. The whole database will be maintained by database administrator. There are varieties of trains where the passengers can select the train according to the convenience for their destination journey. The journey could be within the state or across the India. Each train has the three types of classes i.e. Sleeper class, First class and the AC compartment. Design the application for the above problem description.

**CASE STUDY 5: RECRUITMENT PROCEDURE FOR SOFTWARE INDUSTRY**

**Problem Statement:**

In the software industry the recruitment procedure is the basic thing that goes in the hand with the requirement as specified by the technical management team. HR first gives an advertisement in leading Newspapers, Journals, Weeklies and Websites. The job seekers can apply for it through by Post or by e-mail to the company. The technical skill and the experience of the candidates are reviewed and the short listed candidates are called for the interview.
There may be different rounds for interview like the written test, technical interview, HR interview. After the successful completion of all rounds of interview, the selected candidates names are displayed. Meanwhile HR gives all the details about the salary, working hours, terms and conditions and the retirement benefit to the candidate.

**CASE STUDY 6: DESIGN A STUDENT REGISTRATION SYSTEM**

*Problem Statement:*

Each student has access to his or her course and grade information only and must be authenticated prior to viewing or updating the information. A course instructor will use the system to view the list of courses he or she is assigned for a given semester or has taught previously, view the list of students registered for the course(s) he or she is teaching, and record final grades for each student in the course(s). TA assignments will also be viewable through this system. Instructors must also be authenticated prior to viewing or updating any information.

**CASE STUDY 7: ONLINE AUCTION SALES**

*Problem Statement:*

The online auction system is a design about a website where sellers collect and prepare a list of items they want to sell and place it on the website for visualizing. To accomplish this purpose the user has to access the site. Incase it’s a new user he has to register. Purchaser’s login and select items they want to buy and keep bidding for it. Interacting with the purchasers and sellers through messages does this. There is no need for customer to interact with the sellers because every time the purchasers bid, the details will be updated in the database. The purchaser making the highest bid for an item before the close of the auction is declared as the owner of the item. If the auctioneer or the purchaser doesn’t want to bid for the product then there is fixed cutoff price mentioned for every product. He can pay that amount directly and own the product. The purchaser gets a confirmation of his purchase as an acknowledgement from the website. After the transition by going back to the main menu where he can view other items.
REFERENCE BOOKS:


List of Practicals:

1. Study and Practice on various commands like man, passwd, tty, script, clear, date, cal, cp, mv, ln, rm, unlink, mkdir, rmdir, du, df, mount, umount, find, unmask, ulimit, ps, who, w.

2. Study and Practice on various commands like cat, tail, head, sort, nl, uniq, grep, egrep, fgrep, cut, paste, join, tee, pg, comm, cmp, diff, tr, awk, tar, cpio.

3. a) Write a Shell Program to print all .txt files and .c files.
   b) Write a Shell program to move a set of files to a specified directory.
   c) Write a Shell program to display all the users who are currently logged in after a specified time.
   d) Write a Shell Program to wish the user based on the login time.

4. a) Write a Shell program to pass a message to a group of members, individual member and all.
   b) Write a Shell program to count the number of words in a file.
   c) Write a Shell program to calculate the factorial of a given number.
   d) Write a Shell program to generate Fibonacci series.

5. a) Write a Shell program to print all prime numbers between 1 and n.
   b) Write a Shell program to count no of lines in a text file which starts with a specified letter (Use grep command).
6  a) Simulate `cat` command.  b) Simulate `cp` command.

7  a) Simulate `head` command.  b) Simulate `tail` command.

8  a) Simulate `mv` command.  b) Simulate `nl` command.

9  Write a program to handle the signals like `SIGINT`, `SIGQUIT`, `SIGFPE`.

10 Implement the following IPC forms
    a) `FIFO`  b) `PIPE`

11 Implement `message queue` form of IPC.

12 Implement `shared memory` form of IPC.

13 Write a Socket program to print system date and time (Using TCP/IP).
Data Warehousing Exercises:

1. **Introduction to Informatica Power Center 7.1.1 and Introduction to Oracle 9i**

2. **Adding a Repository**
   a. Create a Source Definition using source connection and import the employee data from source table.
   b. Organize the columns in the table view to the requirement of Data Analysis.
   c. Create a Target Definition using target connection to the target table.
   d. Create tables for transformation and generate SQL to perform transformation.

3. **Mapping**
   a. Perform an ETL on Employees database. Select the employee table as the source and the same as the target and assume connectivity and delimiters as pipe without any specific transformations.
   b. Perform an ETL on Employees database. Select the employee table as the source and the same as the target and assume connectivity and delimiters as pipe using expression transformation, filter transformation, router transformation, aggregator transformation and joiner transformation.
   c. Perform and ETL on Employees database, connect the source and target and then perform debug on the filter transformation mapping.

4. **Lookup**
   a. Using the above mappings perform connected lookup with lookup transformation using natural keys and populate the other keys with default values.
Data Mining Exercises:
Associations

1. Derive associations manually from the following dataset.

<table>
<thead>
<tr>
<th>Outlook</th>
<th>Temperature</th>
<th>Humidity</th>
<th>Windy</th>
<th>Play</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sunny</td>
<td>hot</td>
<td>high</td>
<td>false</td>
<td>no</td>
</tr>
<tr>
<td>Sunny</td>
<td>hot</td>
<td>high</td>
<td>true</td>
<td>no</td>
</tr>
<tr>
<td>Overcast</td>
<td>hot</td>
<td>high</td>
<td>false</td>
<td>yes</td>
</tr>
<tr>
<td>Rainy</td>
<td>mild</td>
<td>high</td>
<td>false</td>
<td>yes</td>
</tr>
<tr>
<td>Rainy</td>
<td>cool</td>
<td>normal</td>
<td>false</td>
<td>yes</td>
</tr>
<tr>
<td>Rainy</td>
<td>cool</td>
<td>normal</td>
<td>true</td>
<td>no</td>
</tr>
<tr>
<td>Overcast</td>
<td>cool</td>
<td>normal</td>
<td>true</td>
<td>yes</td>
</tr>
<tr>
<td>Sunny</td>
<td>mild</td>
<td>high</td>
<td>false</td>
<td>no</td>
</tr>
<tr>
<td>Sunny</td>
<td>cool</td>
<td>normal</td>
<td>false</td>
<td>yes</td>
</tr>
<tr>
<td>Rainy</td>
<td>mild</td>
<td>normal</td>
<td>false</td>
<td>yes</td>
</tr>
<tr>
<td>Sunny</td>
<td>mild</td>
<td>normal</td>
<td>true</td>
<td>yes</td>
</tr>
<tr>
<td>Overcast</td>
<td>mild</td>
<td>high</td>
<td>true</td>
<td>yes</td>
</tr>
<tr>
<td>Overcast</td>
<td>hot</td>
<td>normal</td>
<td>false</td>
<td>yes</td>
</tr>
<tr>
<td>Rainy</td>
<td>mild</td>
<td>high</td>
<td>true</td>
<td>no</td>
</tr>
</tbody>
</table>

Clustering

2. Open Weka and Load the data set editor. Get familiarize with the editor operations.
   a. Load the weather.nominal dataset. Use the filter weka.unsupervised.instance.RemoveWithValues to remove all instances in which the humidity attribute has the value high. To do this, first make the field next to the Choose button show the text RemoveWithValues. Then click on it to get the Generic Object Editor window, and figure out how to change the filter settings appropriately.
   b. Undo the change to the dataset that you just performed, and verify that the data has reverted to its original state.

3. Choosing k-means clustering algorithm for clustering use the Cancer data (.arff) perform clustering with a Euclidean distance function and visually inspect the nature of the clusters.
Classification

4. Choosing an appropriate filter for classification use the Iris data (.arff) perform classification and visualize the classification tree.

5. The glass dataset glass.arff from the U.S. Forensic Science Service contains data on six types of glass. Glass is described by its refractive index and the chemical elements that it contains; the aim is to classify different types of glass based on these features. This dataset is taken from the UCI datasets, which have been collected by the University of California at Irvine and are freely available on the Web. They are often used as a benchmark for comparing data mining algorithms. Find the dataset glass.arff and load it into the Explorer interface. For your own information, answer the following exercises. How many attributes are there in the dataset? What are their names? What is the class attribute? Run the classification algorithm IBk (weka.classifiers.lazy.IBk). Use cross-validation to test its performance.