TKR COLLEGE OF ENGINEERING AND TECHNOLOGY
(AUTONOMOUS)
(Sponsored by TKR Educational Society, Approved by AICTE, Affiliated by JNTUH,
Accredited by NAAC with 'A' Grade)

B.TECH – INFORMATION TECHNOLOGY
Course Structure R-20

SEMESTER III

S.No.	Class	Course Code	Name of the Subject	L	T	P	C
1	HS	CHSM1	Business Economics and Financial Analysis	3	0	0	3
2	BS	CBSM4	Mathematical Foundation of Computer Science	3	0	0	3
3	ES	CESDS1	Data Structures	3	0	0	3
4	PC	C63PC1	Digital Logic Design	3	0	0	3
5	PC	C63PC2	Operating Systems	3	0	0	3
6	PC	C63PC3	Linux Programming	3	0	0	3
7	ES	CESDS2	Data Structures Lab	0	0	2	1
8	PC	C63PC4	Linux/Operating Systems Lab	0	0	4	2
9	MC	MC003	Cultural Activity	0	0	0	Satisfactory
Total Credits				**18**	**0**	**6**	**21**

Mandatory Course: Cultural Activity
The student should participate in culture activity (Music/Dance/Singing/etc.) conducted by the College, student should produce the participation certificate for clearing this course.

SEMESTER IV

S.No.	Class	Course Code	Name of the Subject	L	T	P	C
1	BS	CBSM3	Probability & Statistics	3	0	0	3
2	PC	C64PC1	Computer Organization	3	0	0	3
3	PC	C64PC2	Database Management Systems	3	0	0	3
4	PC	C64PC3	Java Programming	3	0	0	3
5	PC	C64PC4	Design and Analysis of Algorithms	3	0	0	3
6	PC	C64PC5	Formal Languages and Automata Theory	3	0	0	3
7	PC	C64PC6	Java Programming Lab	0	0	3	1.5
8	PC	C64PC7	Database Management Systems Lab	0	0	3	1.5
9	MC	MC004	Video with Social Messages	0	0	0	Satisfactory
Total Credits				**18**	**0**	**6**	**21**

Mandatory Course: Video with Social Messages
Student should make video with social messages. This has to be uploaded in the youtube.com, by maintaining the terms and conditions of youtube.com. Student should produce youtube.com link with screen shot for clearing this mandatory course.

TKR COLLEGE OF ENGINEERING AND TECHNOLOGY
(AUTONOMOUS)
(Sponsored by TKR Educational Society, Approved by AICTE, Affiliated by JNTUH,
Accredited by NAAC with 'A' Grade)

INFORMATION TECHNOLOGY

B.Tech III Semester **L/T/P/C**
3 /0/ 0/ 3

BUSINESS ECONOMICS AND FINANCIAL ANALYSIS (CHSM1)

Course Objectives:

1. To learn the basic business type of the organization.
2. To aquire the knowledge and impact of the economy on business firms.
3. To analyse the business from the financial perspective.
4. To know the financial position of the company.

Course Outcomes:

1. Analyze the total structure of the business and able to identify and classify the different types of business entities.
2. Asses the demand and supply analyses with the help of various measures and types of Elasticity of demand.
3. Infer the knowledge about production and cost analysis for product and services.
4. Interpret the fundamental concepts related to financial accounting.
5. Predict the financial position by analyzing the financial statement of the company through various ratios.

UNIT I
Introduction to Business and Economics

Business: Define Business, characteristics of business, Types of Business Entities, Limited Liability Companies, Sources of Capital for a Company.
Economics: Significance of Economics, Micro and Macro Economic Concepts and Importance of National Income, Business Cycle, Features and Phases of Business Cycle. Nature and Scope of Business Economics, Role of Business Economist.

UNIT II
Demand Analysis

Elasticity of Demand: Elasticity, Types of Elasticity, Law of Demand, Measurement and Significance of Elasticity of Demand, Factors affecting Elasticity of Demand, Elasticity of Demand in decision making, Demand Forecasting: Steps in Demand Forecasting, Methods of Demand Forecasting.

UNIT III
Production, Cost, Market Structures & Pricing

Production Analysis: Production function, Law of returns to scale, Internal and External Economies of Scale. **Cost Analysis:** Cost concepts, Types of costs, Break-Even Analysis (BEA)
Pricing: Types of pricing, product life cycle, **GST (Goods & Service Tax).**

BUSINESS ECONOMICS AND FINANCIAL ANALYSIS (CHSM1)

Market Structures: Types of competition, Features of Perfect competition, Monopoly and Monopolistic competition, oligopoly.

UNIT IV
Financial Accounting

Accounting concepts and Conventions, Accounting Equation, Double-Entry system of Accounting, Rules for maintaining Books of Accounts, Journal, Posting to Ledger, Preparation of Trial Balance, Elements of Financial Statements, and Preparation of Final Accounts.

UNIT V
Financial Analysis through Ratios

Concept of Ratio Analysis, Liquidity Ratios, Turnover Ratios, Profitability Ratios, Proprietary Ratios, Solvency, Leverage Ratios (simple problems).

Text Books:

1. D. D. Chaturvedi, S. L. Gupta, Business Economics - Theory and Applications, International Book House Pvt. Ltd. 2013.
2. Dhanesh K Khatri, Financial Accounting, Tata McGraw Hill, 2011.
3. Geethika Ghosh, Piyali Gosh, Purba Roy Choudhury, Managerial Economics, 2e, Tata McGraw Hill Education Pvt. Ltd. 2012.
4. Rakesh Garg, Sandeep Garg Hand book of GST in India.
5. A.R. Aryasri (2011) Managerial Economics and Financial Analysis, TMH, India.

Reference Books:

1. Paresh Shah, Financial Accounting for Management 2e, Oxford Press, 2015.
2. S. N. Maheshwari, Sunil K Maheshwari, Sharad K Maheshwari, Financial Accounting, 5e,Vikas Publications, 2013.

TKR COLLEGE OF ENGINEERING AND TECHNOLOGY
(AUTONOMOUS)
(Sponsored by TKR Educational Society, Approved by AICTE, Affiliated by JNTUH,
Accredited by NAAC with 'A' Grade)

INFORMATION TECHNOLOGY

B.Tech III Semester **L/T/P/C**
3 /0/ 0/ 3

MATHEMATICAL FOUNDATIONS OF COMPUTER SCIENCE (CBSM4)

Course Objectives:
To learn:

1. To introduce the concepts of of mathematical logic.
2. To describe the importance and limitations of predicate logic.
3. Use division into cases in a proof. Use concept of division and its properties in different cases.
4. To relate practical examples to the appropriate set, function or relation model, and interpret the associated operations and terminology in context.
5. Introduce the concepts of semi groups, monoids, groups, sub-groups, abelian groups.
6. Isomorphism and homomorphism of groups.

Course Outcomes:

After learning the contents of this paper the student must be able to
CO1: Apply mathematical logic to solve problems.
CO2: Analyse the assertions using predicate logic.
CO3: Analyse different properties of GCD.
CO4: Find the GCD using Division and Euclidean Algorithm.
CO5: Illustrate the basic terminology of functions, relations, sets and demonstrate knowledge of their associated operations.
CO6: Understand the importance of algebraic properties with regard to working within various number systems.

UNIT I
Mathematical Logic

Statements and notations, Connectives, well formed formulas, truth tables, tautology, equivalence implication, normal forms, Quantifiers, universal quantifiers.

UNIT II
Predicates

Predicative logic, free and bounded variables, rules of inference, consistency, proof of contradiction.

UNIT III
Principles of Mathematical Induction

The well ordering principle, recursive definition, division algorithm, prime numbers, greatest common divisor, Euclidean algorithm, fundamental theorem of arithmetic.

MATHEMATICAL FOUNDATIONS OF COMPUTER SCIENCE (CBSM4)

UNIT IV
Relations

Properties of Binary relations, equivalence, transitive closure, compatibility and partial ordering Relations, Hasse diagram
Functions: Inverse function, composition of functions, recursive functions.

UNIT V
Groups

Algebraic structures, examples and general properties, Semi groups and monoids, Groups and
Sub groups, cosets and Lagranges theorem, homomorphism, and isomorphism of groups, cyclic groups, permutation groups.

Text Books:

1. Discrete Mathematics for Computer scientists & Mathematicians, J. L. Mott, A. Kandel, T.P.Baker.
2. Discrete mathematics and its Applications, Kenneth H.Rosen, fifth edition. TMH

Reference Books:

1. Elements of Discrete mathematics, C.L.Liu ,D.P.Mohapatra,4th edition,McGraw Hill education (India) Private Limited
2. Discrete mathematical structures theory and applications- malik & Sen Cengage.
3. Discrete mathematics with applications, Thomas Koshy, Elsevier.
4. Logic and Discrete mathematics, grass Man & Trembley, Pearson Education.

TKR COLLEGE OF ENGINEERING AND TECHNOLOGY
(AUTONOMOUS)
(Sponsored by TKR Educational Society, Approved by AICTE, Affiliated by JNTUH,
Accredited by NAAC with 'A' Grade)

INFORMATION TECHNOLOGY

B.Tech III Semester **L/T/P/C**
3 /0/ 0/ 3

DATA STRUCTURES (CESDS1)

Course Objectives:

1. To understand the basic concepts such as Abstract Data Types, Linear and Non Linear Data structures.
2. To understand the notations used to analyze the Performance of algorithms.
3. To understand the behaviour of data structures such as stacks, queues, trees, hash tables, search trees, Graphs and their representations.
4. To choose an appropriate data structure for a specified application.
5. To understand and analyze various searching and sorting algorithms.
6. To learn to implement ADTs such as lists, stacks, queues, trees, graphs, search trees in C++ to solve problems.

Course Outcomes:

1. Understand the concept of ADT.
2. Ability to choose appropriate data structures to represent data items in real world problems.
3. Ability to analyses the time and space complexities of algorithms.
4. Ability to design programs using a variety of data structures such as stacks, queues, hash tables, binary trees, search trees, heaps, graphs, and B-trees.
5. Able to analyze and implement various kinds of searching and sorting techniques.

UNIT I
C++ Programming Concepts

Review of C, input and output in C++, functions in C++-value parameters, reference parameters, Parameter passing, function overloading, function templates, arrays, pointers, new and delete operators, class and object, access specifiers, friend functions, constructors and destructor, class templates, Inheritance and Polymorphism.

UNIT II
Linked List

Linear list ADT-array representation and linked representation, Singly Linked Lists-Operations-Insertion, Deletion, Circularly linked lists-Operations for Circularly linked lists, Doubly Linked Lists- Operations-Insertion, Deletion.
Stack and Queues: Stack ADT, definition, array and linked list implementations, applications-infix to postfix conversion, Postfix expression evaluation, Queue ADT, definition, array and linked list Implementations.

UNIT III
Trees

Definition, terminology, Binary trees-definition, Properties of Binary Trees, Binary Tree ADT, representation of Binary Trees-array and linked representations, Binary Tree traversals.

Heap Tree: Max Heap-Definition, Insertion into a Max Heap, Deletion from a Max Heap.

DATA STRUCTURES (CESDS1)

UNIT IV
Hashing

Introduction, Hashing functions, Collision Techniques- Linear Probing, Quadratic Probing, Double Hashing.

Sorting: Insertion Sort, Selection Sort, Quick sort, Heap Sort, Merge sort, Comparison of Sorting methods.

UNIT V
Graphs

Definitions, Terminology, Applications and more definitions, Properties, Graph ADT, Graph Representations- Adjacency matrix, Adjacency lists, Graph Search methods - DFS and BFS.

Binary Search Trees: Definition, Representation and Operations- Searching, Insertion and Deletion.

Balanced Search Trees: AVL Trees, Red- Black Trees, B- Trees.

Text Books:

1. Data structures, Algorithms and Applications in C++, 2nd Edition, SartajSahni, Universities Press.
2. Data structures and Algorithms in C++, Adam Drozdek, 4th edition, Cengage learning.

Reference Books:

1. Data structures with C++, J. Hubbard, Schaum's outlines, TMH.
2. Data structures and Algorithms in C++, M.T. Goodrich, R. Tamassia and D. Mount, Wiley India.
3. Data structures and Algorithm Analysis in C++, 3rd edition, M. A. Weiss, Pearson.
4. Classic Data Structures, D. Samanta, 2nd edition, PHI.

TKR COLLEGE OF ENGINEERING AND TECHNOLOGY
(AUTONOMOUS)
(Sponsored by TKR Educational Society, Approved by AICTE, Affiliated by JNTUH,
Accredited by NAAC with 'A' Grade)

INFORMATION TECHNOLOGY

B.Tech III Semester **L/T/P/C**
3 /0/ 0/ 3

DIGITAL LOGIC DESIGN (C63PC1)

Course Objectives:

1. To learn basic techniques for the design of digital circuits and fundamental concepts used in the design of digital systems.
2. To understand common forms of number representation in digital electronic circuits and to be able to convert between different representations.
3. To implement simple logical operations using combinational logic circuits.
4. To impart to student the concepts of sequential circuits, enabling them to analyze sequential systems in terms of state machines.
5. To implement synchronous state machines using flip-flops.
6. To implement memory devices using RAM and ROM.

Course Outcomes:

Upon completion of the Course, the students will be able to
CO1: Convert numeric information in different forms, e.g. different bases, signed integers, various codes such as ASCII, Gray, and BCD.
CO2: Realize simple Boolean expressions using the theorems and postulates of Boolean algebra and to minimize combinational functions.
CO3: Design and analyze of small combinational circuits and to use standard combinational functions/building blocks to build larger more complex circuits.
CO4: Design of sequential logic circuits and synthesizing of threshold functions.
CO5: Design and analyze small sequential circuits and devices and to use standard sequential functions/building blocks to build larger more complex circuits.

UNIT I
Number System and Switching Functions

Number Systems, Base Conversion Methods, Complements of Numbers, Codes- Binary Codes, Binary Coded Decimal Code and its Properties, Unit Distance Codes, Error Detecting and Correcting Codes.

UNIT II
Boolean Algebra

Basic Theorems and Properties, Switching Functions, Canonical and Standard Form, Algebraic Simplification of Digital Logic Gates, Properties of XOR Gates, Universal Gates, Multilevel NAND/NOR realizations.
Minimization of Boolean Functions: Introduction, The Minimization with theorem, The Karnaugh Map Method, Five and Six Variable Maps, Prime and Essential Implications, Don't Care Map conditions, Tabular Method.

DIGITAL LOGIC DESIGN (C63PC1)

UNIT III
Combinational Circuits

Introduction, Adders, Subtractors, Multiplexers, Demultiplexers, Encoders, Decoders, code converters, Comparators and Hazards.

UNIT IV
Sequential Circuits I

Introduction, Basic Differences between Combinational and Sequential circuits, the Binary Cell, Latch, Flip-Flop-Types, and Race around condition, Excitation tables and characteristic equations. Conversion from one type of Flip-Flop to another, preset and Clear inputs, Timing and Triggering Consideration, Clock Skew.
Sequential Circuits II: Introduction, Register-Types, Counter –Types, Design of Ripple (mod-N) Counter, Ring Counter.

UNIT V
Sequential Machines

State Diagram, Analysis of Synchronous Sequential Circuits, Approaches to the Design of Synchronous Sequential Finite State Machines, Finite state machine- capabilities and limitations, Mealy and Moore models-minimization of completely specified and incompletely specified sequential machines, Partition techniques and Merger chart methods-concept of minimal cover table.

Text Books:

1. Digital Design- Morris Mano, PHI, 3rd Edition.
2. Switching and Finite Automata Theory- Zvi Kohavi & Niraj K. Jha, 3rd Edition, Cambridge.

References Books:

1. Anand Kumar, “Switching Theory and Logic Design” PHI, 2008.
2. Charles H. Roth, “Fundamentals of Logic Design” Thomson Publications, 5th Edition, 2004.

TKR COLLEGE OF ENGINEERING AND TECHNOLOGY
(AUTONOMOUS)
(Sponsored by TKR Educational Society, Approved by AICTE, Affiliated by JNTUH,
Accredited by NAAC with 'A' Grade)

INFORMATION TECHNOLOGY

B.Tech III Semester **L/T/P/C**
3 /0/ 0/ 3

OPERATING SYSTEMS (C63PC2)

Course Objectives:

To gain insight knowledge on performance and working of an operating system.

Course Outcomes:

The student will be able to

1. Able to understand the basic overview of operating systems and system calls
2. Ability to solve synchronization problem with Two-Process solution, Petersons solutions and apply the concepts of minimization of turnaround time, waiting time and response time to find CPU scheduling Problems
3. Apply the page replacement algorithms to identify the page fault in the given string.
4. Able to distinguish between file access methods and allocation methods.
5. Ability to apply Bankers Algorithm to avoid deadlocks and change access controls to protect files.

UNIT I
Overview

Introduction-Operating system objectives, User view, System view, Operating system definition, Computer System Architecture, OS Structure, OS Operations, Process Management, Memory Management, Storage Management, Computing Environments. Operating System services, User and OS Interface, System Calls, Types of System Calls, System Programs, Operating System Design and Implementation.

UNIT II
CPU Scheduling Process

Concepts-The Process, Process State, Process Control Block, Threads, Process Scheduling-Scheduling Queues, Schedulers, Context Switch, Operations on Processes, Inter-process communication-ordinary pipes and named pipes, message queues.
Process Scheduling: Basic concepts, Scheduling Criteria, Scheduling algorithms, Multiple-Processor Scheduling, Real-Time Scheduling, Thread scheduling. Process Synchronization, Background, The Critical Section Problem, Peterson's solution, Synchronization Hardware, Semaphores, Classic Problems of Synchronization, Monitors.

UNIT III
Memory Management

Memory Management Strategies- Background, Swapping, Contiguous Memory Allocation, Segmentation, Paging, Structure of Page Table.
Virtual Memory Management: Background, Demand Paging, Copy-on-Write, Page Replacement, Page Replacement Algorithms, Allocation of Frames, Thrashing, Virtual memory in Windows.

OPERATING SYSTEMS (C63PC2)

UNIT IV
Storage Management

File System- Concept of a File, System calls for file operations - open (), read (), write (), close (), seek (), unlink (), Access methods, Directory and Disk Structure, File System Mounting, File Sharing.
File System Implementation: File System Structure, File System Implementation, Directory Implementation, Allocation methods, Free-space Management, Efficiency, and Performance. Overview of Mass Storage Structure.

UNIT V
Deadlocks

System Model, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, and Recovery from Deadlock.
Protection: System Protection, Goals of Protection, Principles of Protection, Domain of Protection, Access Matrix, Implementation of Access Matrix, Access Control, Revocation of Access Rights, Capability-Based Systems, Language-Based Protection.

Text Books:

1. Operating System Concepts , Abraham Silberschatz, Peter B. Galvin, Greg Gagne, 8th Edition, Wiley, 2016 India Edition
2. Operating Systems – Internals and Design Principles, W. Stallings, 7th Edition, Pearson.

Reference Books:

1. Modern Operating Systems, Andrew S Tanenbaum, 3rd Edition, PHI
2. Operating Systems: A concept-based Approach, 2nd Edition, D.M. Dhamdhere, TMH.
3. Principles of Operating Systems, B. L. Stuart, Cengage learning, India Edition.
4. An Introduction to Operating Systems, P.C.P. Bhatt, PHI.
5. Principles of Operating systems, Naresh Chauhan, Oxford University Press.

TKR COLLEGE OF ENGINEERING AND TECHNOLOGY
(AUTONOMOUS)
(Sponsored by TKR Educational Society, Approved by AICTE, Affiliated by JNTUH,
Accredited by NAAC with 'A' Grade)

INFORMATION TECHNOLOGY

B.Tech III Semester **L/T/P/C**
3 /0/ 0/ 3

LINUX PROGRAMMING (C63PC3)

Course Objectives:

1. To understand and make effective use of Linux utilities and Shell scripting language (bash) to solve Problems.
2. To implement in C some standard Linux utilities such as ls, mv, cp etc. using system calls.
3. To develop the skills necessary for systems programming including file system programming, process and signal management, and interprocess communication.
4. To develop the basic skills required to write network programs using Sockets.

Course Outcomes:

1. Work confidently in Linux environment.
2. Work with shell script to automate different tasks as Linux administration.

UNIT I
Linux Utilities

File handling utilities, Security by file permissions, Process utilities, Disk utilities, Networking commands, Filters, Text processing utilities and Backup utilities. Sed-Scripts, Operation, Addresses, Commands, Applications, awk-Execution, Fields and Records, Scripts, Operation, Patterns, Actions, Associative Arrays, String and Mathematical functions, System commands in awk, Applications.

Shell programming with Bourne again shell (bash) - Introduction, shell responsibilities, pipes and Redirection, here documents, running a shell script, the shell as a programming language, shell meta characters, file name substitution, shell variables, command substitution, shell commands, the environment, quoting, test command, control structures, arithmetic in shell, shell script examples, interrupt processing, functions, debugging shell scripts.

UNIT II
Files and Directories

File Concept, File types, File System Structure, file metadata-Inodes, kernel support for files, system calls for file I/O operations- open, creat, read, write, close, lseek, dup2, file status information-stat family, file and record locking- fcntl function, file permissions - chmod, fchmod, file ownership-chown, lchown, fchown, links-soft links and hard links – symlink, link, unlink.

Directories - Creating, removing and changing Directories-mkdir, rmdir, chdir, obtaining current working directory-getcwd, Directory contents, Scanning Directories-opendir, readdir, closedir, rewinddir functions.

LINUX PROGRAMMING (C63PC3)

UNIT III
Process

Process concept, Layout of a C program image in main memory, Process environment- environment list, environment variables, getenv, setenv, Kernel support for process, process identification, process control - process creation, replacing a process image, waiting for a process, process termination, zombie process, orphan process, system call interface for process management-fork, vfork, exit, wait, waitpid, exec family, Process Groups, Sessions and Controlling Terminal, Differences between threads and processes.
Signals – Introduction to signals, Signal generation and handling, Kernel support for signals, Signal function, unreliable signals, reliable signals, kill, raise, alarm, pause, abort, sleep functions.

UNIT IV
Inter Process Communication

Introduction to IPC, IPC between processes on a single computer system, IPC between processes on different systems, pipes-creation, IPC between related processes using unnamed pipes, FIFOs-creation, IPC between unrelated processes using FIFOs (Named pipes), differences between unnamed and named pipes, popen and pclose library functions. **Message Queues** - Kernel support for messages, APIs for message queues, client/server example. **Semaphores** - Kernel support for semaphores, APIs for semaphores, file locking with semaphores.

UNIT V
Shared Memory

Kernel support for shared memory, APIs for shared memory, shared memory example. **Sockets** - Introduction to Berkeley Sockets, IPC over a network, Client- Server model, Socket address structures (Unix domain and Internet domain), Socket system calls for connection oriented protocol and connectionless protocol, example-client/server programs-Single Server-Client connection, Multiple simultaneous clients, Socket optionssetsockopt and fcntl system calls.

Text Books:

1. Unix System Programming using C++, T. Chan, PHI.
2. Unix Concepts and Applications, 4th Edition, Sumitabha Das, TMH.
3. Unix Network Programming, W. R. Stevens, PHI.

Reference Books:

1. Beginning Linux Programming, 4th Edition, N. Matthew, R. Stones, Wrox, Wiley India Edition.
2. Unix for programmers and users, 3rd Edition, Graham Glass, King Ables, Pearson.
3. System Programming with C and Unix, A. Hoover, Pearson.
4. Unix System Programming, Communication, Concurrency and Threads, K. A. Robbins and S. Robbins, Pearson Education.
5. Unix shell Programming, S. G. Kochan and P. Wood, 3rd edition, Pearson Education.
6. Shell Scripting, S. Parker, Wiley India Pvt. Ltd.
7. Advanced Programming in the Unix Environment, 2nd edition, W. R. Stevens and S.A. Rago, Pearson Education.
8. Unix and Shell programming, B. A. Forouzan and R. F. Gilberg, Cengage Learning.
9. Linux System Programming, Robert Love, O'Reilly, SPD.
10. C Programming Language, Kernighan and Ritchie, PHI.

TKR COLLEGE OF ENGINEERING AND TECHNOLOGY
(AUTONOMOUS)
(Sponsored by TKR Educational Society, Approved by AICTE, Affiliated by JNTUH,
Accredited by NAAC with 'A' Grade)

INFORMATION TECHNOLOGY

B.Tech III Semester **L/T/P/C**
0 /0/ 2/ 1

DATA STRUCTURES LAB (CESDS2)

Course Objectives:

1. To understand basic concepts of data structures and abstract data types.
2. To write and execute programs in C++ to solve problems using data structures such as arrays, linked lists, stacks, queues, trees, graphs, hash tables and search trees.
3. To learn to write C++programs to implement various sorting algorithms.

Course Outcomes:

1. Able to identify the appropriate data structures and algorithms for solving real world problems.
2. Able to implement various kinds of searching and sorting techniques.
3. Able to implement data structures such as stacks, queues, Search trees, and hash tables to solve various computing problems.

LIST OF EXPERIMENTS:

1. Write a C++ Program to implement Stack using Array.

2. Write a C++ Program to Postfix evaluation.

3. Write a C++ Program to implement Queue using Array.

4. Write a C++ Program to implement Single linked list operations.

5. Program to implement Double linked list operations.

6. Program to implement Circular linked list operations.

7. Program to implement Stack using Linked list.

8. Program to implement Queue using Linked list.

9. Program to implement sorting techniques.

a) Insertion Sort b) Selection Sort c) Quick sort d) Heap Sort e) Merge sort

Reference Books:

1. Data Structures using C++, D. S. Malik, 2nd edition, Cengage learning.
2. Data Structures using C++, V. Patil, Oxford University Press.
3. Fundamentals of Data structures in C++, 2nd edition, E. Horowitz, S. Sahni and D. Mehta, Universities Press.
4. C++ Data Structures, 4th edition, Nell Dale, Jones and Bartlett student edition.

TKR COLLEGE OF ENGINEERING AND TECHNOLOGY
(AUTONOMOUS)
(Sponsored by TKR Educational Society, Approved by AICTE, Affiliated by JNTUH,
Accredited by NAAC with 'A' Grade)

INFORMATION TECHNOLOGY

B.Tech III Semester **L/T/P/C**
0 /0/ 4/ 2

LINUX/OPERATING SYSTEMS LAB (C63PC4)

Course Objectives:

1. To write shell scripts to solve problems.
2. To implement some standard Linux utilities such as ls,cp etc using system calls.
3. To understand the operating System functionalities

Course Outcomes:

1. 1. Ability to understand the Linux environment
2. Ability to perform the file management and multiple tasks using shell scripts in Linux Environment.
3. Able to implement various Scheduling algorithms.
4. Able to detect and solve deadlocks.

List of problems:

Note: Use Bash for Shell scripts.

1. Write a shell script that accepts a file name, starting and ending line numbers as arguments and displays all the lines between the given line numbers.
2. Write a shell script that deletes all lines containing a specified word in one or more files supplied as arguments to it.
3. Write a shell script that displays a list of all the files in the current directory to which the user has read, write and execute permissions.
4. Write a shell script that receives any number of file names as arguments checks if every argument supplied is a file or a directory and reports accordingly. Whenever the argument is a file, the number of lines on it is also reported.
5. Write a shell script that accepts a list of file names as its arguments, counts and reports the occurrence of each word that is present in the first argument file on other argument files.
6. Write a shell script to list all of the directory files in a directory.
7. Write a shell script to find factorial of a given integer.
8. Write an awk script to count the number of lines in a file that do not contain vowels.
9. Write an awk script to find the number of characters, words and lines in a file.
10. Implement in C the following Linux commands using System calls a) cat b) mv

OPERATING SYSTEMS LAB

List of Programs:

1. Simulate the following CPU scheduling algorithms
   a) Round Robin b) SJF c) FCFS d) Priority
2. Simulate all file allocation strategies
   a) Sequential b) Indexed
3. Simulate MVT and MFT
4. Simulate all File Organization Techniques
   a) Single level directory b) Two level c) Hierarchical
5. Simulate Bankers Algorithm for Dead Lock Avoidance
6. Simulate Bankers Algorithm for Dead Lock Prevention
7. Simulate all page replacement algorithms
   a) FIFO b) LRU.

Reference Books:

1. Beginning Linux Programming, 4th Edition, N. Matthew, R. Stones, Wrox, Wiley India Edition.
2. Advanced Unix Programming, N. B. Venkateswarulu, BS Publications.
3. Unix and Shell Programming, M.G. Venkatesh Murthy, Pearson Education.
4. Unix Shells by Example, 4th Edition, Elllie Quigley, Pearson Education.

TKR COLLEGE OF ENGINEERING AND TECHNOLOGY
(AUTONOMOUS)
(Sponsored by TKR Educational Society, Approved by AICTE, Affiliated by JNTUH,
Accredited by NAAC with 'A' Grade)

INFORMATION TECHNOLOGY

B.Tech IV Semester **L/T/P/C**
3 /0/ 0/ 3

PROBABILITY & STATISTICS (CBSM3)

Course Objectives:
To learn:

1. Random variables that describe randomness or an uncertainty in certain realistic situation.
2. The study of discrete and continuous distribution predominantly describes important probability distribution.
3. Sampling distribution of mean, variance, point estimation and interval estimation.
4. The testing of Hypothesis of Large samples
5. The testing of Hypothesis of small samples
6. The basic ideas of statistics including correlation and regression

Course Outcomes:

After learning the contents of this paper the student must be able to learn the concept of

CO1: Random variables and various discrete and continuous probability distributions and their properties.
CO2: Calculate interval estimations of Mean and Proportion of large samples.
CO3: Make important decisions s for few samples which are taken from a large data.
CO4: Calculate Mean and Proportion and to make important decisions from large samples which are taken from normal populations.
CO5: Test the hypothesis and give the inference to the given data.
CO6: The statistical methods of studying data sample.

UNIT I
Random Variables & Distributions

Random variables –discrete and continuous, mass function, density function of probability distributions, Binomial, Poisson and Normal distributions related properties.

UNIT II
Sampling Distributions

Sampling distributions of means (σknown and unknown).Estimation theory, point estimations, Interval estimations, Maximum Error.

UNIT III
Large Samples

Null hypothesis, alternative hypothesis, Type-1, Type-2 errors, critical region, confidence interval for mean, testing of single mean and two means, confidence interval for the proportions ,Test of Hypothesis for the single proportions and difference between the proportions.

PROBABILITY & STATISTICS (CBSM3)

UNIT IV
Small Samples

Small samples Test for single mean, difference of two means, test for ratio of variances, Chi-square test for goodness of fit and independence of attributes.

UNIT V
Basic Statistics

Correlation and regression, Rank correlation, Curve fitting by the method of least squares-fitting of straight lines, second degree parabolas, power and exponential curves.

Text Books:

1. Probability & Statistics for Engineers by G.S.S. BhismaRao, SciTech Publications.
2. Probability & Statistics for Engineers by D.K.Murugesan & P.Guru Swamy, Anuradha Publications.

Reference Books:

1. W.Feller- An introduction to probability theory and its applications- Vol.1- 3rd edition Wiley-1968.
2. Probability & Statistics for Engineers, Millers and John E.Freund, Prentice Hall of Ind.

TKR COLLEGE OF ENGINEERING AND TECHNOLOGY
(AUTONOMOUS)
(Sponsored by TKR Educational Society, Approved by AICTE, Affiliated by JNTUH,
Accredited by NAAC with 'A' Grade)

INFORMATION TECHNOLOGY

B.Tech IV Semester **L/T/P/C**
3 /0/ 0/ 3

COMPUTER ORGANIZATION (C64PC1)

Course Objectives:

1. To understand basic components of computers.
2. To understand the I/O organization.
3. To understand the Memory Organization.
4. To understand the Architecture of 8086.
5. To understand the instruction formats and representation of data at the machine level and how computations are performed.

Course Outcomes:

1. Able to understand the basic components and the design of CPU, ALU and Control Unit.
2. Ability to understand the data transfer between I/O devices.
3. Ability to understand Memory hierarchy and its impact on computer cost/performance.
4. Ability to use instruction sets and formats of 8086.
5. Able to write assembly language programs to solve problems.

UNIT I
Basic Computer Organization - Functions of CPU, I/O Units, Memory Instruction

Instruction Formats - One address, two addresses, zero addresses and three addresses and comparison, registers, Instructions timing and control, instruction cycle, addressing modes with numeric examples: Program Control - Status bit conditions, conditional branch instructions, Program Interrupts: Interrupt cycle, Types of Interrupts, control memory Micro programmed address sequencing, micro program control unit example and micro instruction format, RISC, CISC-Processors.

UNIT II
Input-Output Organizations - I/O Interface, I/O Bus and Interface Modules

I/O Vs memory Bus, Isolated Vs Memory-Mapped I/O, Asynchronous data Transfer-Strobe Control, Hand Shaking: Asynchronous Serial transfer- Asynchronous Communication interface, Modes of transfer programmed I/O, Interrupt Initiated I/O, DMA; DMA Controller, DMA Transfer, IOP-CPU-IOP Communication.

COMPUTER ORGANIZATION (C64PC1)

UNIT III
Memory Organizations

Memory hierarchy, Main Memory, RAM, ROM Chips, Memory Address Map, Memory Connection to CPU, associate memory, Cache Memory, Data Cache, Instruction cache, Miss and Hit ratio, Access time associative, set associative, mapping, waiting into cache, Introduction to virtual memory.

UNIT IV

8086 CPU PIN Diagram - Architectural features, special and general purpose registers, segment registers, 8086 Flag registers, Addressing modes of 8086, operating modes of 8086, Interrupt types, memory segmentations.

UNIT V
8086-Instruction Formats

Assembly Language Programs involving branch & Call instructions, sorting, evaluation of arithmetic expressions.

Text Books:

1. Computer System Architecture: Moris Mano (UNIT - 1, 2, 3).
2. Advanced Micro Processor and Peripherals - Hall/ A K Ray (UNIT - 4, 5).

Reference Books:

1. Computer Organization and Architecture – William Stallings Sixth Edition, Pearson/PHI.
2. Structured Computer Organization and Design - Andrew S. Tanenbaum, 4th Edition PHI/Pearson.
3. Fundamentals or Computer Organization and Design - Sivaraama Dandamudi Springer Int. Edition.

TKR COLLEGE OF ENGINEERING AND TECHNOLOGY
(AUTONOMOUS)
(Sponsored by TKR Educational Society, Approved by AICTE, Affiliated by JNTUH,
Accredited by NAAC with 'A' Grade)

INFORMATION TECHNOLOGY

B.Tech IV Semester **L/T/P/C**
3 /0/ 0/ 3

DATABASE MANAGEMENT SYSTEMS (C64PC2)

Course Objectives:

1. To understand the basic concepts and the applications of database systems.
2. To master the basics of SQL and construct queries using SQL.
3. To understand the relational database design principles.
4. To become familiar with the basic issues of transaction processing and concurrency control.
5. To become familiar with database storage structures and access techniques.

Course Outcomes:

1. Demonstrate the basic elements of a relational database management system.
2. Ability to identify the data models for relevant problems.
3. Ability to design entity relationship model and convert entity relationship diagrams into RDBMS and formulate SQL queries on the data.
4. Apply normalization for the development of application software.
5. Understand transaction processing, concurrency control and recovery techniques.

UNIT I
Introduction

Database System Applications, Purpose of Database Systems, View of Data, Database Languages – DDL, DML, Database Design, Data Storage and Querying, Transaction Management, Database Architecture, Database Users and Administrators, History of Database Systems.
Introduction to Database design: Database Design and ER diagrams, Entities, Attributes and Entity sets, Relationships and Relationship sets, Additional features of ER Model, Conceptual Design with the ER Model.

UNIT II
Relational Model

Introduction to the Relational Model, Integrity Constraints over Relations, Enforcing Integrity constraints, Querying relational data, Logical data base Design: ER to Relational, Introduction to Views, Destroying /Altering Tables and Views. **Relational Algebra and Calculus:** Relational Algebra, Relational calculus – Tuple relational Calculus, Domain relational calculus. SQL Queries, Constraints, Form of Basic SQL Query, UNION, INTERSECT and EXCEPT, Nested Queries, Aggregate Operators, NULL values Complex Integrity Constraints in SQL, Triggers and Active Data bases, Designing Active Databases.

DATABASE MANAGEMENT SYSTEMS (C64PC2)

UNIT III
Schema Refinement and Normal Forms

Introduction to Schema Refinement, Functional Dependencies - Reasoning about FDs, Normal Forms - ***1NF, 2NF, 3NF***, Properties of Decompositions, Normalization, Schema Refinement in Database Design, Other Kinds of Dependencies.

UNIT IV
Transaction Management

Transaction Atomicity and Durability, Transaction Isolation, Serializability, Transaction Isolation and Atomicity, Implementation of Isolation Levels.
Concurrency Control: Lock–Based Protocols, Multiple Granularity, Timestamp-Based Protocols, Validation-Based Protocols, Multiversion Schemes.

UNIT V
Recovery System

Failure Classification, Storage, Recovery and Atomicity, Recovery Algorithm, Buffer Management, Failure with loss of nonvolatile storage, Early Lock Release and Logical Undo Operations, Remote Backup systems.
Storage and Indexing: Overview of Storage and Indexing: Data on External Storage, File Organization and Indexing, Index Data Structures, comparison of File Organizations. Tree-Structured Indexing: Intuition for tree Indexes, Indexed Sequential Access Method (ISAM).

Text Books:

1. Data base Management Systems, Raghu Ramakrishnan, Johannes Gehrke, McGraw Hill Education (India) Private Limited, 3rd Edition.
2. Data base System Concepts, A. Silberschatz, Henry. F. Korth, S. Sudarshan, McGraw Hill Education(India) Private Limited l, 6th edition.

Reference Books:

1. Database Systems, 6th edition, R Elmasri, ShamkantB.Navathe, Pearson Education.
2. Database System Concepts, Peter Rob & Carlos Coronel, Cengage Learning.
3. Introduction to Database Management, M. L. Gillenson and others, Wiley Student Edition.
4. Database Development and Management, Lee Chao, Auerbach publications, Taylor & Francis Group.

TKR COLLEGE OF ENGINEERING AND TECHNOLOGY
(AUTONOMOUS)
(Sponsored by TKR Educational Society, Approved by AICTE, Affiliated by JNTUH,
Accredited by NAAC with 'A' Grade)

INFORMATION TECHNOLOGY

B.Tech IV Semester **L/T/P/C**
3 /0/ 0/ 3

JAVA PROGRAMMING (C64PC3)

Course Objectives:

1. To introduce the object oriented programming concepts.
2. To understand object oriented programming concepts, and apply them in solving problems.
3. To introduce the principles of inheritance and polymorphism; and demonstrate how they relate to the design of abstract classes.
4. To introduce the implementation of packages and interfaces.
5. To introduce the concepts of exception handling and multithreading.
6. To introduce the design of Graphical User Interface using applets and swing controls.

Course Outcomes:

1. Able to understand the use of inheritance and abstract classes.
2. Able to gain knowledge on how to use packages, interfaces, I/O stream classes.
3. Able to handle exceptions by using exceptional handling mechanisms.
4. Able to develop multithreaded applications with synchronization.
5. Able to solve problems using java collection framework.
6. Able to develop applets for web applications and design GUI based applications

UNIT I
Object-Oriented Thinking

History of object-oriented programming, overview of java, Object oriented design, Structure of java program, Java buzzwords, Data types, Variables and Arrays, operators, expressions, control statements, Access specifiers, Introducing classes, Methods and Constructors, String handling.
Inheritance– Inheritance concept, Inheritance basics, Member access, Constructors, Creating Multilevel hierarchy, benefits of inheritance, costs of inheritance, super uses, using final with inheritance, method over loading ,method overriding, abstract classes, Object class, Polymorphism.

UNIT II
Packages

Defining a Package, Creating a package, Access protection, importing packages.
Interfaces- defining an interface, implementing interfaces, Nested interfaces, applying interfaces, variables in interfaces and extending interfaces.
Stream based I/O (java.io) – File class, The Stream classes-Byte streams and Character streams, Print Writer , The Console class, Serialization, De-Serialization, Enumerations, Wrapper Classes.

JAVA PROGRAMMING (C64PC3)

UNIT III
Exception Handling

Fundamentals of exception handling, Exception hierarchy, Exception types, Uncaught exceptions, using try and catch, multiple catch clauses, nested try statements, throw, throws and finally, built- in exceptions, creating own exception.
Multithreading- Differences between thread-based multitasking and process-based multitasking, Java thread model, creating threads(lifecycle),Thread methods, creating multiple threads**,** thread priorities, synchronizing threads, inter thread communication.

UNIT IV
The Collections Framework (java.util)

Limitations of arrays, Collections overview, Collection Interfaces, The Collection classes- ArrayList, LinkedList, Stack, Vector, Cursors, HashSet, TreeSet, Comparable, comparators, map interfaces and Classes, Queue – priority queue, collections.

UNIT V
GUI Programming with Swing

Introduction, limitations of AWT, MVC architecture, components, containers. Understanding Layout Managers, Flow Layout, Border Layout, Grid Layout, Card Layout, Grid Bag Layout.
Event Handling- The Delegation event model- Events, Event sources, Event Listeners, Event classes, Handling mouse and keyboard events, Adapter classes.
Applets – Applets and HTML, Security Issues, Applets and Applications, passing parameters to applets. Creating a Swing Applet, Painting in Swing, A Paint example,

Text Books:

1. Java The complete reference, 9th edition, Herbert Schildt, McGraw Hill Education (India) Pvt. Ltd.
2. Understanding Object-Oriented Programming with Java, updated edition, T. Budd, Pearson Education.

Reference Books:

1. An Introduction to programming and OO design using Java, J. Nino and F.A. Hosch, John Wiley & sons.
2. Introduction to Java programming, Y. Daniel Liang, Pearson Education.
3. Object Oriented Programming through Java, P. Radha Krishna, Universities Press.
4. Programming in Java, S. Malhotra, S. Chudhary, 2nd edition, Oxford Univ. Press.
5. Java Programming and Object oriented Application Development, R. A. Johnson, Cengage Learning.

TKR COLLEGE OF ENGINEERING AND TECHNOLOGY
(AUTONOMOUS)
(Sponsored by TKR Educational Society, Approved by AICTE, Affiliated by JNTUH,
Accredited by NAAC with 'A' Grade)

INFORMATION TECHNOLOGY

B.Tech IV Semester **L/T/P/C**
3 /0/ 0/ 3

DESIGN AND ANALYSIS OF ALGORITHMS (C64PC4)

Course Objectives:

1. To analyze performance of algorithms.
2. To choose the appropriate data structure and algorithm design method for a specified application.
3. To understand how the choice of data structures and algorithm design methods impacts the performance of programs.
4. To solve problems using algorithm design methods such as the greedy method, divide and conquer, dynamic programming, backtracking and branch and bound.
5. To understand the differences between tractable and intractable problems.
6. To introduce P and NP classes.

Course Outcomes:

1. Able to analyze the performance of the algorithm in terms of time and space & apply the concept of divide & conquer method on various examples.
2. Able to find out the solution for the given example problems by using Backtracking & apply the concept of graph problems on various examples.
3. Able to solve the problems with Greedy method for the given example problems.
4. Able to solve optimization problems using Dynamic Programming.
5. Able to solve the given example problems using Branch & Bound and design the deterministic & non deterministic algorithms and categorize them as a Np-hard and Np-complete problems accordingly.

UNIT I
Introduction

Introduction: Algorithm definition, Algorithm specification, Performance analysis.
Divide and conquer- General method, applications - Binary search, Merge sort, Quick sort, Strassen's Matrix Multiplication.

UNIT II
Disjoint Set Operations

Disjoint set operations, union and find algorithms, AND/OR graphs, Graph Traversals, Connected Components and Spanning trees, Bi-connected components **Backtracking**-General method, applications- The 4-queen problem, The 8-queen problem, sum of subsets problem, graph coloring, Hamiltonian cycles.

DESIGN AND ANALYSIS OF ALGORITHMS (C64PC4)

UNIT III
Greedy Method

General method, applications- Knapsack problem, Job sequencing with deadlines, Minimum cost spanning trees, Single source shortest path problem.

UNIT IV
Dynamic Programming

General Method, applications- Chained matrix multiplication, All pairs shortest path problem, Optimal binary search trees, 0/1 knapsack problem, Reliability design, Traveling sales person problem.

UNIT V
Branch and Bound

General Method, applications-0/1 Knapsack problem, LC Branch and Bound solution, FIFO Branch and Bound solution, Traveling sales person problem.
NP-Hard and NP-Complete problems- Basic concepts, Non-deterministic algorithms, NP - Hard and NP-Complete classes, Cook's theorem.

Text Books:

1. Fundamentals of Computer Algorithms, 2nd Edition, Ellis Horowitz, Sartaj Sahni and S. Rajasekharan, Universities Press.
2. Design and Analysis of Algorithms, P. H. Dave, H. B. Dave, 2nd edition, Pearson Education.

Reference Books:

1. Algorithm Design: Foundations, Analysis and Internet examples, M. T. Goodrich and R. Tomassia, John Wiley and sons.
2. Design and Analysis of Algorithms, S. Sridhar, Oxford Univ. Press
3. Design and Analysis of algorithms, Aho, Ullman and Hopcroft, Pearson Education.
4. Foundations of Algorithms,, R. Neapolitan and K. Naimipour, 4th edition, Jones and Bartlett Student edition.
5. Introduction to Algorithms, 3rd Edition, T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, PHI.

TKR COLLEGE OF ENGINEERING AND TECHNOLOGY
(AUTONOMOUS)
(Sponsored by TKR Educational Society, Approved by AICTE, Affiliated by JNTUH,
Accredited by NAAC with 'A' Grade)

INFORMATION TECHNOLOGY

B.Tech IV Semester **L/T/P/C**
3 /0/ 0/ 3

FORMAL LANGUAGES AND AUTOMATA THEORY (C64PC5)

Course Objectives:

To provide introduction to some of the central ideas of theoretical computer science from the perspective of formal languages.

Course Outcomes:

The Student will be able to

1. Demonstrate the abstract models of computing includes Finite Automata.
2. Construct FA for regular expressions and vice versa and minimization &equivalence of FA.
3. Design CFG for the given Formal language.
4. Design PDA for CFG and prove the equivalence of CFG and PDA.
5. Design Normal forms for CFG and design a TM for a given language.
6. Distinguish between Decidable and Undecidable problems.

UNIT I
Introduction

The Central Concepts of Automata Theory – Symbol, Alphabets, Strings, Language. Finite Automata. Finite Automata without output - Deterministic Finite Automata, Nondeterministic Finite Automata, Finite Automata with Epsilon-Transitions, Finite automata with output - Mealy and Moore machines, Equivalence of Mealy and Moore machines.

UNIT II
Regular Expressions

Finite Automata and Regular Expressions, Applications of Regular Expressions, Algebraic Laws for Regular Expressions, Converting FA's to Regular Expressions, Converting Regular Expressions to FA. Pumping Lemma for Regular Languages, Applications of the Pumping Lemma, Closure Properties of Regular Languages, Decision Properties of Regular Languages, Equivalence and Minimization of Finite Automata.

UNIT III
Context-Free Grammars

Definition of Context-Free Grammars, Derivations Using a Grammar, Sentential Applications of Context-Free Grammars, Leftmost and Rightmost Derivations Ambiguity in Grammars and Languages, Minimization of CFG-Removal of useless symbols, Elimination of epsilon productions, Removal of unit productions.

FORMAL LANGUAGES AND AUTOMATA THEORY (C64PC5)

Push Down Automata: Definition of the Pushdown Automata, the Languages of a PDA, Instantaneous description of PDA, Equivalence of PDA's and CFG's, Deterministic Pushdown Automata, non-deterministic pushdown automata

UNIT IV
Normal Forms for Context- Free Grammars

Chomsky's normal form(CNF), Greibach normal form(GNF), the Pumping Lemma for Context-Free Languages, Closure Properties of Context-Free Languages. Decision Properties of CFL's.

Introduction to Turing Machines:
Definition of TM, Instantaneous description of TM, Design of TM's, Programming Techniques for Turing Machines, Extensions to the basic Turing machine. Universal Turing machine.

UNIT V
Decidability and Undecidability

Decidable and undecidable problems, Recursive language and properties of RE, Recursive Enumerable language and properties of REL, The Post Correspondence Problem, Undecidability of the PCP, Tractable Intractable Problems, The Classes P and NP,NP-complete, NP-hard problems.

Text Books:

1. Introduction to Automata Theory, Languages, and Computation, 3nd Edition, John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman, Pearson Education.
2. Introduction to the Theory of Computation, Michael Sipser, 3rd edition, Cengage Learning.

Reference Books:

1. Introduction to Languages and The Theory of Computation, John C Martin, TMH.
2. Introduction to Computer Theory, Daniel I.A. Cohen, John Wiley.
3. A Text book on Automata Theory, P. K. Srimani, Nasir S. F. B, Cambridge University Press.
4. Introduction to Formal languages Automata Theory and Computation Kamala Krithivasan, Rama R, Pearson.
5. Theory of Computer Science – Automata languages and computation, Mishra and Chandrashekaran, 2nd edition, PHI.

TKR COLLEGE OF ENGINEERING AND TECHNOLOGY
(AUTONOMOUS)
(Sponsored by TKR Educational Society, Approved by AICTE, Affiliated by JNTUH,
Accredited by NAAC with 'A' Grade)

INFORMATION TECHNOLOGY

B.Tech IV Semester **L/T/P/C**
0/0/3/1.5

JAVA PROGRAMMING LAB (C64PC6)

Course Objectives:

1. To write programs using abstract classes.
2. To write programs for solving real world problems using java collection frame work.
3. To write multithreaded programs.
4. To write GUI programs using swing controls in Java.
5. To introduce java compiler and eclipse platform.

Course Outcomes:

1. Able to write programs for solving real world problems using java collection frame work.
2. Able to write programs using abstract classes.
3. Design and develop programs using objects and inheritance in Java language.
4. Able to write multithreaded programs.
5. Able to write GUI programs using swing controls in Java.

Note:

1. Use Linux and MySQL for the Lab Experiments. Though not mandatory, encourage the use of Eclipse platform.
2. The list suggests the minimum program set. Hence, the concerned staff is requested to add more problems to the list as needed.

List of Programs

1. a) Write a JAVA program to display the Fibonacci sequence
   b) Write a JAVA program to check whether given string is palindrome or not.
   c) Write a JAVA program to give the example for 'this' keyword.
2. a) Write a JAVA program to demonstrate static variables, methods, and blocks.
   b) Write a JAVA program to demonstrate 'super' keyword.
   c) Write a JAVA program that illustrates simple inheritance.
3. a) Write a JAVA program that illustrates multi-level inheritance
   b) Write a JAVA program to demonstrate method overloading and method overriding.
   c) Write a JAVA program demonstrating the difference between method overloading and constructor overloading.
   d) Write a java program to demonstrate abstract classes.
4. a) Write a java program to demonstrate access specifiers.
   b) Write a java program to import packages.

JAVA PROGRAMMING LAB (C64PC6)

5.
   a) Write a java program to demonstrate interfaces.
   b) Write a java program to illustrate nested interfaces.
   c) Write a java program to demonstrate extending interfaces.
6.
   a) Write a java program to demonstrate Byte streams and character streams.
   b) Write a java program to demonstrate PrintWriter class.
   c) Write a java program to demonstrate Serialization and De-Serialization.
   d) Write a java program to demonstrate Enumeration and Wrapper classes.

7.
   a) Write a java program to demonstrate try, catch, finally keywords.
   b) Write a java program to demonstrate throw and throws keyword.
8.
   a) Write a java program to demonstrate multiple catch clauses and nested try statements.
   b) Write a java program to demonstrate creating own exception.
9.
   a) Write a simple java program how to create a thread.
   b) Write a java program to demonstrate thread priorities.
   c) Write a java program to demonstrate synchronization in threads.
   d) Write a java program to demonstrate inter thread communication.
10.
   a) Write a java program to perform linear search in a given collection.
   b) Write a java program to perform sorting in a given collection.
11.
   a) Write a java program to demonstrate Cursors.
   b) Write a java program to demonstrate Comparable and Comparator.
12.
   a) Write a java program to demonstrate Map classes.
   b) Write a java program to demonstrate priority queue.

13. Write a java program to demonstrate
   a) Flow Layout
   b) Border Layout
   c) Grid Layout
   d) Card Layout
   e) Grid Bag Layout.
14. Write a java program to demonstrate mouse events and keyboard events.
15.
   a) Write a simple Applet program to print “Hello World.”
   b) Write a java program to create simple Swing Applet.

Text Books:

1. Java The complete reference, 9th edition, Herbert Schildt, McGraw Hill Education (India) Pvt. Ltd.
2. Understanding Object-Oriented Programming with Java, updated edition, T. Budd, Pearson Education.

TKR COLLEGE OF ENGINEERING AND TECHNOLOGY
(AUTONOMOUS)
(Sponsored by TKR Educational Society, Approved by AICTE, Affiliated by JNTUH,
Accredited by NAAC with 'A' Grade)

INFORMATION TECHNOLOGY

B.Tech IV Semester **L/T/P/C**
0/0/3/1.5

DATABASE MANAGEMENT SYSTEMS LAB (C64PC7)

Course Objectives:

This lab enables the students to practice the concepts learnt in the subject DBMS by developing a database for an example company named "Roadway Travels" whose description is as follows. The student is expected to practice the designing, developing and querying a database in the context of example database "Roadway travels". Students are expected to use "Mysql" database.

Course Outcomes:

1. Formulate queries using SQL DML/DDL/DCL commands.
2. Analyze the normalization techniques
3. Design and implement a database schema for given problem.
4. Develop programs using PL/SQL

Roadway Travels

"Roadway Travels" is in business since 1997 with several buses connecting different places in India. Its main office is located in Hyderabad. The company wants to computerize its operations in the following areas:

Reservations & Cancellation

Reservations are directly handled by booking office. Reservations can be made 30 days in advance and tickets issued to passenger. One Passenger/person can book many tickets (to his/her family). Cancellations are also directly handed at the booking office.

In the process of computerization of Roadway Travels you have to design and develop a Database which consists the data of Buses, Passengers, Tickets, and Reservation and cancellation details. You should also develop query's using SQL to retrieve the data from the database.

The above process involves many steps like 1. Analyzing the problem and identifying the Entities and Relationships, 2. E-R Model 3. Relational Model 4. Normalization 5.

Creating the database 6. Querying. Students are supposed to work on these steps week wise and finally create a complete "Database System" to Roadway Travels. Examples are given at every experiment for guidance to students.

Experiment 1: E-R Model

Analyze the problem carefully and come up with the entities in it. Identify what data has to be persisted in the database. This contains the entities, attributes etc. Identify the primary keys for all the entities. Identify the other keys like candidate keys, partial keys, if any.

Example: Entities: 1. Bus 2. Ticket 3. Passenger Relationships 1. Reservation. 2. Cancellation.

Primary Key Attributes: 1. Ticket_ID (Ticket Entity) 2. Passenger_ID (Passenger Entity)

3. Bus_No. (Bus Entity)

Apart from the above mentioned entities you can identify more. The above mentioned are few.

Note: The student is required to submit a document by writing the Entities and Keys to the lab teacher.

DATABASE MANAGEMENT SYSTEMS LAB (C64PC7)

Experiment 2: Concept design with E-R Model

Relate the entities appropriately. Apply cardinalities for each relationship. Identify strong entities and weak entities (if any). Indicate the type of relationships (total / partial). Try to incorporate generalization, aggregation, specialization etc wherever required. Example: E-R diagram for bus

Note: The student is required to submit a document by drawing the E-R Diagram to the lab teacher.

Experiment 3: Relational Model

Represent all the entities (Strong, Weak) in tabular fashion. Represent relationships in a tabular fashion. There are different ways of representing relationships as tables based on the cardinality. Represent attributes as columns in tables or as tables based on the requirement. Different types of attributes (Composite, Multi-valued, and Derived) have different way of representation.

Example: The passenger tables look as below. This is an example. You can add more attributes based on your E-R model. This is not a normalized table.

Passenger, Name, Age, Sex, Address, Passenger_ID. Ticket_ID

Note: The student is required to submit a document by Represent relationships in a tabular fashion to the lab teacher.

Experiment 4: Normalization

Database normalization is a technique for designing relational database tables to minimize duplication of information and, in so doing, to safeguard the database against certain types of logical or structural problems, namely data anomalies. For example, when multiple instances of a given piece of information occur in a table, the possibility exists that these instances will not be kept consistent when the data within the table is updated, leading to a loss of data integrity. A table that is sufficiently normalized is less vulnerable to problems of this kind, because its structure reflects the basic assumptions for when multiple instances of the same information should be represented by a single instance only. For the above table in the First normalization we can remove the multi valued attribute Ticket_id and place it in another table along with the primary key of passenger.

First Normal Form: The above table can be divided into two tables as shown below.Passenger, Name, Age, Sex, Address, Passenger_ID, Ticket_ID.You can do the second and third normal forms if required.

Experiment 5: Installation of Mysql and practicing DDL commands

Installation of MySql. In this week you will learn Creating databases, How to create tables, altering the database, dropping tables and databases if not required. You will also try truncate, rename commands etc.

Example for creation of a normalized “Passenger” table.

CREATE TABLE Passenger (Passport_id INTEGER PRIMARY KEY, Name VARCHAR (50) Not NULL, Age Integer Not NULL, Sex Char, Address VARCHAR (50) Not NULL);

Similarly create all other tables.

Note: Detailed creation of tables is given at the end.

Experiment 6: Practicing DML commands

DML commands are used to for managing data within schema objects. Some examples: (UNDERSTANDING) SIMPLE

SELECT - retrieve data from the a database

INSERT - insert data into a table

UPDATE - updates existing data within a table

DATABASE MANAGEMENT SYSTEMS LAB (C64PC7)

DELETE - deletes all records from a table, the space for the records remain
Inserting values into "Bus" table:
Insert into Bus values (1234,'hyderabad', 'tirupathi');
Insert into Bus values (2345,'hyderabd','Banglore');
Insert into Bus values (23,'hyderabd','Kolkata');
Insert into Bus values (45,'Tirupathi,'Banglore');
Insert into Bus values (34,'hyderabd','Chennai');
Inserting values into "Passenger" table:
Insert into Passenger values (1, 45,'ramesh', 45,'M','abc123');
Insert into Passenger values (2, 78,'geetha', 36,'F','abc124');
Insert into Passenger values (45, 90,'ram', 30,'M','abc12');
Insert into Passenger values (67, 89,'ravi', 50,'M','abc14');
Insert into Passenger values (56, 22,'seetha', 32,'F','abc55');
Few more Examples of DML commands:
Select * from Bus; (selects all the attributes and display)
UPDATE BUS SET Bus No = 1 WHERE BUS NO=2;

Experiment 7: Querying

In this week you are going to practice queries (along with sub queries) using ANY, ALL, IN, Exists, NOT EXISTS, UNION, INTERSECT, Constraints etc.

Practice the following Queries:

1. Display unique PNR_No of all passengers.
2. Display all the names of male passengers.
3. Display the ticket numbers and names of all the passengers.
4. Find the ticket numbers of the passengers whose name start with 'r' and ends with 'h'.
5. Find the names of passengers whose age is between 30 and 45.
6. Display all the passengers names beginning with 'A'
7. Display the sorted list of passengers names

Experiment 8: Aggregate functions

You are going to practice queries using Aggregate functions (COUNT, SUM, AVG, and MAX and MIN), GROUP BY, HAVING and Creation and dropping of Views.

Experiment 9: Querying using clauses

Write a Query to display the Information present in the Passenger and cancellation tables. Hint: Use UNION Operator.

1. Display the number of days in a week on which the 9W01 bus is available.
2. Find number of tickets booked for each PNR_no using GROUP BY CLAUSE. Hint: Use GROUP BY on PNR_No.
3. Find the distinct PNR numbers that are present.
4. Find the number of tickets booked by a passenger where the number of seats is greater than 1.
   Hint: Use GROUP BY, WHERE and HAVING CLAUSES.
5. Find the total number of cancelled seats.

Experiment 10: PL/SQL

1. Create a PL/SQL block for Addition of Two Numbers
2. Develop a PL/SQL block for IF Condition

DATABASE MANAGEMENT SYSTEMS LAB (C64PC7)

3. Design a PL/SQL block for IF and else condition
4. Develop a PL/SQL block for greatest of three numbers using IF AND ELSEIF
5. Construct a PL/SQL block for summation of odd numbers using for LOOP.

Experiment 11 : Triggers

In this week you are going to work on Triggers. Creation of insert trigger, delete trigger, update trigger. Practice triggers using the above database.

Eg: CREATE TRIGGER updcheck BEFORE UPDATE ON passenger
FOR EACH ROW
BEGIN
IF NEW.TickentNO > 60 THEN SET New.Tickent no = Ticket no; ELSE
SET New.Ticketno = 0;
END IF; END;

Tables

1. BUS
   Bus No: Varchar: PK (public key) Source : Varchar Destination : Varchar
2. Passenger
   PPNO: Varchar (15)) : PK Name: Varchar (15) Age : int (4) Sex:Char (10) : Male / Female Address: VarChar (20)
3. Passenger_Tickets
   PPNO: Varchar (15)) : PK Ticket_No: Numeric (9)
4. Reservation
   PNR_No: Numeric (9):FK Journey_date: datetime(8) No_of_seats: int (8)Address: Varchar (50) Contact_No: Numeric (9) --> Should not be less than 9 and Should not accept any other character other than Integer Status: Char (2) : Yes / No
5. Cancellation
   PNR_No: Numeric(9) : FK Journey_date : datetime(8) No_of_seats : int (8)Address : Varchar (50) Contact_No: Numeric (9) --> Should not be less than 9 and Should not accept any other character other than Integer Status: Char (2) : Yes / No
6. Ticket
   Ticket_No: Numeric (9): PK Journey_date : datetime(8) Age : int (4) Sex:Char(10): Male / Female Source : Varchar Destination : Varchar Dep_time : Varchar

Reference Books:

1. Introduction to SQL, Rick F. Vander Lans, Pearson education.
2. Oracle PL/SQL, B. Rosenzweig and E. Silvestrova, Pearson education.
3. SQL & PL/SQL for Oracle 10 g, Black Book, Dr. P. S. Deshpande, Dream Tech.
4. Oracle Database 11 g PL/SQL Programming, M. Mc Laughlin, TMH.