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

B.Tech - Computer Science and Engineering
Course Structure R-22

SEMESTER III

S.No	Course Classificati	Course Code	Name of the subject	L	T	P	C	I	E	Total
01	**BS**	D3BSPS1	Probability and Statistics	3	1	0	4	40	60	100
02	**ES**	D3ESSD1	Semiconductor Devices and Circuits	3	0	0	3	40	60	100
03	**ES**	D3ESPP2	Python Programming	2	0	0	2	40	60	100
04	**PC**	D53PC1	Mathematical Foundations of Computer Science	3	0	0	3	40	60	100
05	**PC**	D53PC2	Computer Architecture and Organization	3	0	0	3	40	60	100
06	**PC**	D53PC3	Computer Networks	3	0	0	3	40	60	100
07	**ES**	D3ESPP5	Python Programming Lab	0	0	2	1	40	60	100
08	**ES**	D3ESSD2	Semiconductor Devices and Circuits Lab	0	0	2	1	40	60	100
TOTAL				**17**	**1**	**4**	**20**	**320**	**480**	**800**

SEMESTER IV

S.No.	Course Classification	Course Code	Name of the subject	L	T	P	C	I	E	Total
01	**PC**	D54PC4	Object Oriented Programming through Java	3	0	0	3	40	60	100
02	**PC**	D54PC5	Design and Analysis of Algorithms	3	1	0	4	40	60	100
03	**PC**	D54PC6	Database Management Systems	3	0	0	3	40	60	100
04	**PC**	D54PC7	Software Engineering	3	0	0	3	40	60	100
05	**PC**	D54PC8	Web Technologies	2	0	0	2	40	60	100
06	**PC**	D54PC9	Database Management Systems Lab	0	0	2	1	40	60	100
07	**PC**	D54PC10	Object Oriented Programming through Java Lab	0	0	2	1	40	60	100
08	**HS**	D4HSE3	Advanced English Communication Skills Lab	0	1	2	2	40	60	100
09	**PC**	D54PC11	Web Technologies Lab	0	0	2	1	40	60	100
10	**MC**	MC002	Constitution of India*	3	0	0	0	0	0	S
TOTAL				**17**	**2**	**8**	**20**	**360**	**540**	**900**
Mandatory Course: Constitution of India										

COMPUTER SCIENCE AND ENGINEERING

B.Tech III Semester **L/T/P/C**
3/1/0/4

PROBABILITY AND STATISTICS (D3BSPS1)

Pre-requisites: Mathematics courses of first year of study

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:

1. After learning the contents of this the student must be able to learn the concept of
2. Random variables and various discrete and continuous probability distribution and their properties.
3. Calculate interval estimations of Mean and Proportion of large samples.
4. Make important decisions s for few samples which are taken from a large data.
5. Calculate Mean and Proportion and to make important decisions from large samples which are taken from normalpopulations.
6. Test the hypothesis and give the inference to the given data.
7. The statistical methods of studying data sample.

UNIT I
Random variables

Discrete and continuous Random variables- properties- Expectation of discrete and continuous Random variables and Variance of a sum.

UNIT II
Probability Distributions

Binomial- poison and Normal-evaluation of statistical parameters for these three distributions and related problems.

UNIT III
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.

UNIT IV

Sampling distributions, Estimations and Hypothesis-I

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

UNIT V

Testing of hypothesis – II

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, ANOVA.

TEXT BOOKS

1. S C Gupta and V K Kapoor, Fundamentals of Mathematical statistics, Khanna publications
2. Probability and Statistics by Dr. T.K.V. Iyengar, B. Krishna Gandhi, S.Chand publications
3. Miller and Freunds Probability and Statistics for Engineers, Pearson publications

REFERENCE BOOKS

1. B.S. Grewal- Higher Engineering Mathematics- Khanna Publishers- 40th Edition- 2015
2. S.Ross- A First Course in Probability- 6th Ed.- Pearson publications
3. Sheldon M Ross, Probability and statistics for Engineers and scientists, academic press

COMPUTER SCIENCE AND ENGINEERING

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

SEMICONDUCTOR DEVICES AND CIRCUITS (D3ESSD1)

Pre-requisite: Fundamentals of Semiconductors.

Course Objectives:

1. To introduce the concepts of various semiconductor devices like Diodes, Transistors, FET's and MOSFET'S.
2. To impart the knowledge of various configurations, characteristics and applications of electronic circuits.

Course Outcomes:

After this course, the student will be able to

1. Acquire the knowledge in semiconductor materials and knows the complete internal structure of PN junction its capacitances, resistances
2. Acquire the knowledge of different types of diodes including their modes of operation, etc.
3. Design the circuits for the conversion of AC to DC Voltages.
4. Gain knowledge in the structure of Transistor, FET,MOS (different types, operation, characteristics and applications)
5. Acquire knowledge in the dc bias circuitry of BJT.

UNIT I

Introduction to semiconductor physics: Formation of P-type and N-type semiconductors, principle and operation of Diode, Volt-Ampere characteristics, Current components in a p-n Diode, Diode equation, Temperature dependence, Static and dynamic resistances, Equivalent circuit, Load line analysis, Diffusion and Transition Capacitances.

UNIT II

Special Purpose Devices: Breakdown Mechanisms in Semi-Conductor Diodes, Zener diode characteristics, Use of Zener diode as simple regulator, Principle of operation and Characteristics of Tunnel Diode (With help of Energy band diagram) and Varactor Diode, Principle of Operation of SCR, UJT operation and its characteristics.

UNIT III

Rectifiers and Filters: P-N junction diode as a Rectifier - Half Wave Rectifier, Ripple Factor, Full Wave Rectifier, Bridge Rectifier, Harmonic components in Rectifier Circuits, Filters(Qualitative treatment)

UNIT IV

Bipolar Junction Transistor (BJT): Construction, Principle of Operation, Transistor current components, Transistor as an Amplifier, Common Emitter, Common Base and Common Collector configurations and their Characteristics. Operating point, DC Load line, Transistor Biasing techniques (Qualitative treatment)

UNIT V

Field Effect Transistor (FET & MOSFET): Construction, Principle of Operation of FET and MOSFET, Pinch-Off Voltage, Volt-Ampere Characteristic, Comparison of BJT and FET, MOSFET.

Text Books:

1. Electronic Devices and Circuits – R.L. Boylestad and Louis Nashelsky, PEI/PHI, 9th Ed, 2006.
2. Millman’s Electronic Devices and Circuits – J. Millman and C. C. Halkias, Satyabratajit, TMH, 2/e, 1998.

References:

1. Electronic Devices and Circuits –S.Salivahanan, NSuresh Kumar & A.Vallavaraj
2. Electronic Devices and Circuits – Sanjeev Gupta & Santosh Gupta, Dhanpat Rai Publications

COMPUTER SCIENCE AND ENGINEERING

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

PYTHON PROGRAMMING (D3ESPP2)

Course Objective:

To understand and learn the concepts of basic python programming, as it is a current programming constructs used for real time applications.

Course Outcomes:

After completion of course the student will be able to

1. Understand the basic concepts of python programming-L2
2. Illustrate operators, conditional statements, loops in python –L3
3. Construct code and test small python programs using functions and data structures –L3
4. Develop different programs using file concept modules of python – L5
5. Apply the concepts of object – oriented programming in python- L3

UNIT I
INTRODUCTION

Introduction to Python, History, Need of Python Programming, features Applications, python environment setup, Basic syntax, Variables, Data Types, Keywords, Input-Output, Indentation, script structure, Running Python Scripts.

UNIT II
OPERATORS

Arithmetic Operators, Comparison (Relational) Operators, Assignment Operators, Logical Operators, Bitwise Operators, Membership Operators, Identity Operators, Expressions and order of evaluations, Conditional statements if, if-else Looping Control Structures for, while Control Statements: Break, Continue, Pass.

UNIT III
FUNCTIONS

Defining Functions, Calling Functions, Passing Arguments, Keyword Arguments, Default Arguments, Variable-length arguments, Anonymous Functions, Fruitful Functions (Function Returning Values), Scope of the Variables in a Function - Global and Local Variables.

DATA STRUCTURES

Lists, Tuples, dictionaries, sets, Sequences, Comprehensions.

UNIT IV

FILES

File input/output, Text processing file functions.

MODULES

Creating modules, import statement, from. Name spacing, Packages, using packages, implementing packages: numpy, pandas, Django framework, iterator tools, scipy, matplot lib.

UNIT V

Object Oriented Programming in Python

Classes, 'self-variable', Methods, Constructor Method, Inheritance, Overriding Methods, Data hiding.

Error and Exception Handling

Difference between an error and Exception, Handling Exception, try except block, Raising Exceptions, User Defined Exceptions.

Text Books:

1. Python Programming: A Modern Approach, Vamsi Kurama, Pearson.
2. Wesley J. Chun “Core Python Programming”, Second Edition, Prentice Hall.

Reference Books:

1. Allen Downey, “Think Python”, Second Edition, Green Tea Press.
2. Introduction to Computation & Programming Using Python, Spring 203 Edition, By John V.Guttag.
3. Programming in Python A Complete Introduction to the Python Language (Developer's Library), by Mark Summerfield, 2ndEdition

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

COMPUTER SCIENCE AND ENGINEERING

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

MATHEMATICAL FOUNDATIONS OF COMPUTER SCIENCE (D53PC1)

Course Objectives:
To learn:

1. To introduce the concepts 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 course, the student must be able to

1. Apply mathematical logic to solve problems.
2. Analyze the assertions using predicate logic.
3. Analyze different properties of GCD.
4. Find the GCD using Division and Euclidean Algorithm.
5. Illustrate the basic terminology of functions, relations, sets and demonstrate knowledge of their associated operations.
6. 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.

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 NBA & NAAC with 'A' Grade)

COMPUTER SCIENCE AND ENGINEERING

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

COMPUTER ARCHITECTURE AND ORGANIZATION (D53PC2)

Course Objective:

Understand the detailed computer architecture and organization, hardware operation of digital computer.

Course Outcomes:

After learning the contents of this course, the student will be able to

1. Understand the basic computer organization and design L2
2. Design the hardwired and micro-programmed control units and demonstrate 8086 architecture L3
3. Analyze the computer arithmetic operations and write 8086 basic ALP programs L4
4. Analyze I/O data transfer modes and memory hierarchy. L4
5. Analyze the concurrent processing L4

UNIT I

Digital Computers

Introduction, Block diagram of Digital Computer, Definition of Computer Organization, Computer Design and Computer Architecture .Basic Computer Organization and Design Instruction codes, Computer Registers, Computer instructions, Timing and Control, Instruction cycle, Memory Reference Instructions, Input – Output and Interrupt, Complete Computer Description.

UNIT II

Central Processing Unit

Processor Organization, Register Organization, Instruction cycle, hardwired control unit, Micro program control unit. The 8086 Processor Architecture, Register organization, Physical memory organization, General Bus Operation, I/O Addressing Capability, Special Processor Activities, Minimum and Maximum mode system and timings.

UNIT III

Computer Arithmetic

Introduction, The arithmetic logic unit, Integer representation, Integer arithmetic, Floating point representation, Floating point arithmetic, Addition and Subtraction, Multiplication Algorithms, Division Algorithms, Floating - point Arithmetic operations. Data Transfer and Manipulation Instructions and 8086 ALP basic programs.

UNIT IV

Input-Output Organization

Peripheral Devices, Input Output Interface, Asynchronous data transfer, Modes of Transfer, Priority Interrupt, Direct memory Access, Input–Output Processor (IOP), Intel 8089 IOP.

Memory Organization Memory Hierarchy, Auxiliary memory, Associate Memory, Cache Memory, Virtual Memory, Memory Management Hardware.

UNIT V

Pipeline and Vector Processing

Parallel Processing, Pipelining, Arithmetic Pipeline, Instruction Pipeline, RISC Pipeline, CISC, RISC versus CISC, Vector Processing, Array Processors.

Multi Processors

Characteristics of Multi processors, Inter connection Structures, Inter processor arbitration, Inter processor communication, and synchronization.

Text Books:

1. Computer System Architecture, M.Moris Mano, Third Edition, Pearson.
2. Advanced Micro processors and Peripherals, KM Bhurchandi, A.K Ray, 3rdedition, McGraw Hill India Education Private Ltd.

Reference Books:

1. Microprocessors and Interfacing, DV Hall, SSSP Rao, 3rdedition, McGraw Hill India Education Private Ltd.
2. Carl Hamacher, Zvonko Vranesic, Safwat Zaky: Computer Organization, 5th Edition, Tata McGraw Hill, 2002.
3. Computer Organization and Architecture, William Stallings, 9th Edition, Pearson.
4. David A. Patterson, John L. Hennessy: Computer Organization and Design–The Hardware /Software Interface ARM Edition,4th Edition

COMPUTER SCIENCE AND ENGINEERING

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

COMPUTER NETWORKS (D53PC3)

Course Objective:
Identify the components required to build different types of networks and choose the required functionality at each layer for a given application

Course Outcomes:
Upon completion of the course the student will be able:

1. To understand the protocol layering and physical level communication.
2. To analyze the performance of a network.
3. To understand the various components required to build different networks.
4. To learn the functions of the network layer and the various routing protocols.
5. To understand the functions and protocols of the Transport layer.

UNIT I
Introduction: OSI, TCP/IP, and other network models, Examples of Networks: Novell Networks, Arpanet, Internet, Network Topologies WAN, LAN, MAN.
Physical Layer: Transmission media copper, twisted pair wireless, switching and encoding asynchronous communications; Narrowband, broadband ISDN and ATM.

UNIT II
Data link layer: Design issues, framing, error detection and correction, CRC, Elementary Protocol- stop and wait, Sliding Window, Slip, Data link layer in HDLC, Internet, ATM.
Medium Access sub layer: ALOHA, MAC addresses, Carrier sense multiple access. IEEE 80.X Standard Ethernet, wireless LANs. Bridges.

UNIT III
Network Layer: Virtual circuit and Datagram subnets-Routing algorithm shortest path routing, Flooding, Hierarchical routing, Broadcast, Multicast, distance vector routing. Dynamic routing – Broadcast routing. Rotary for mobility, The Network layer on the internet and in the ATM Networks.

UNIT IV
Transport Layer: Transport Services, Connection management, TCP and UDP protocols; Congestion Control Algorithms – General Principles – of Congestion prevention policies ATM AAL Layer Protocol.

UNIT V

Application Layer – Domain name system, SNMP, Electronic Mail (SMTP, POP3, IMAP, MIME) the World WEB, HTTP.

Text Books:

1. Computer Networks — Andrew S Tanenbaum, 4th Edition. Pearson Education/PHI.

Reference Books:

1. An Engineering Approach to Computer Networks-S.Keshav, 2nd Edition, Pearson Education.
2. Understanding communications and Networks, 3rd Edition, W.A. Shay, Thomson.
3. Data Communications and Networking – Behrouz A. Forouzan. Third Edition TMH

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

COMPUTER SCIENCE AND ENGINEERING

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

PYTHON PROGRAMMING LAB (D3ESPP5)

Course Objective:

Write and execute the programs based on operators, functions, simple data structures, basic packages using python programming constructs.

Course Outcomes:

After completion of course the student will be able to

1. Use fundamental programming elements : operators ,statements, conditional and control flow statements-L3
2. Compare & contrast predefined functions and build functions- L4
3. Solve various computing problems using python modules and data structures –L4
4. Apply oops concepts using python –L3

List of programs

1. Write a python program to print "Hello World".
2. Running instructions in Interactive interpreter and a Python Script.
3. Write a Python Programming to demonstrate the Indentation.
4. Write a Python program to calculate number of days between two dates.
5. Write a python program that takes 2 numbers as command line arguments and prints its product.
6. Write a Python program to test whether a given letter is a vowel or not.
7. Write a Python program to create a pattern.

```
*
**
***
****
*****
```

8. Write a Python program to count the number 6 in a given list.
9. Write a python program to find the sum of the first n positive integers.
10. Write a Python program to calculate the sum of the digits in an integer
11. Write a Python program that prints all the numbers from 0 to 50 except multiples of 10
12. Write a Python program to check if a number is positive, negative or zero.
13. Write a Python program that will accept the base and height of a triangle and compute the area.
14. Write a Python program to compute the greatest common divisor (GCD) of two positive integers.

15. Write a Python program Make a Simple Calculator

16. Write a Python program to count the number of even and odd numbers from a series of numbers.
17. Write a Python function to calculate the factorial of a number (anon-negative integer).The function should accept the number as an argument.
18. Write a Python function that accepts a string and calculate the number of upper case letters and lower case letters.
19. Write a Python function that checks whether a passed string is palindrome or not.
20. Write a Python program to get the Fibonacci series between 0 to 50usingrecursion
21. Write a Python program to calculate the value of 'a' to the power 'b' using recursion.
22. Write a Python program to get the factorial of a non-negative integer using recursion
23. Write a Python program to calculate the length of a string.
24. Write a Python program to count occurrences of a substring in a string.
25. Write a Python program to count and display the vowels of a given text.
26. Write a program to count the numbers of characters in the string and store them in a dictionary data structure
27. Write a program to use split and join methods in the string and trace a birthday with a dictionary data structure.
28. Write a program combine lists that combines these lists into a dictionary.
29. Write a Python program for binary search.
30. Write a Python program to sort a list of elements using the bubble sort algorithm
31. Write a Python program to sort a list of elements using the quick sort algorithm.
32. Write a Python program to count the frequency of words in a file.
33. Write a Python program to print last n lines of a file.
34. Write a Python program to combine each line from first file with the corresponding line in second file.
35. Write a Python program to assess if a file is closed or not.
36. Write a Python program to get the Python version you are using.
37. Write a Python program to display the current date and time.
38. Write a Python program to print the calendar of a given month and year.
39. Write a Python class which has two methods get_String and print_String.get_String accept a string from the user and print_String print the string in upper case.
40. Write a Python class named Rectangle constructed by length and width and a method which will compute the area of a rectangle.
41. Solve the following linear equations using scipy library
    X+3y+5z=10
    2x+5y+z=8 2x+3y+8z=3
42. Find the determinant for a 2 * 2 matrix using scipy library module.
43. Find the mean and variance for the following data using scipy
    [2, 23, 45, 56, 78, 89, 13, 33, 66, 89]
44. Draw a bar chart with the following data using matplot lib Men_mean=[20,35,30,35,27] Women_mean=[25,32,34,20,25], Men_std=[2,3,4,1,2] Women_std=[3,5,2,3,3]
45. Using matplot lib and scipy libraries, apply the following operations on an image.
    a. Display the image crop image
    b. flip
    c. rotate
    d. Display the statistical information of the image
    e. Turn upside down

COMPUTER SCIENCE AND ENGINEERING

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

SEMICONDUCTOR DEVICES AND CIRCUITS LAB (D3ESSD2)

Course Objectives

1. To impart the knowledge of various configurations, characteristics and applications of various electronic devices.
2. Acquires the knowledge of various biasing circuits of Transistor.

Course Outcomes

After the completion of the lab student can be able to:

1. Design different electronic circuits for different applications using devices like Diodes, Transistors, etc.
2. Design circuits which can convert AC to DC.
3. Design various transistor biasing circuits.
4. Design different Amplifier Circuits.

List of Experiments:

PART A: Electronic workshop practice (in 3 lab sessions):

1. Familiarization of R,L,C, Components
2. Study and operation of
    - Multimeters (Analog and Digital)
    - Function Generator
    - Regulated Power Supplies
    - CRO.
3. Familiarization of Devices, Diodes, BJT, and JFET, MOSFET's

PART B:

1. Forward & Reverse Bias Characteristics of PN Junction. Diode.
2. Zener Diode V-I Characteristics
3. Half Wave Rectifier with & without filters.
4. Full Wave Rectifier with & without filters.
5. Input & Output Characteristics of Transistor in CB Configuration.
6. Input & Output Characteristics of Transistor in CE Configuration.
7. FET Characteristics in CS Configuration.
8. Design of Fixed Bias Circuit.
9. Design of Self Bias Circuit.
10. Transistor as an Amplifier.

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

COMPUTER SCIENCE AND ENGINEERING

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

OBJECT ORIENTED PROGRAMMING THROUGH JAVA (D54PC4)

Course Objective:
Understand the features of object oriented paradigm using JAVA programming

Course Outcomes:
After learning the contents of this course, the student will be able to

1. Demonstrate the concepts of object oriented programming L3
2. Develop programs using java packages, interfaces and stream based I/O. L5
3. Analyze Handling of errors and concurrency using JAVA. L3
4. Analyze the collection frameworks L3
5. Develop applets for web applications and GUI based applications L5

UNIT I

Object-Oriented Thinking

A way of viewing world – Agents and Communities, messages and methods, Responsibilities, Classes and Instances, Class Hierarchies- Inheritance, Method binding, Overriding and Exceptions, Summary of Object-Oriented concepts. 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, Introducing classes, Methods and Classes, String handling.

Inheritance: Inheritance concept, Inheritance basics, Member access, Constructors, Creating Multilevel hierarchy, super uses, using final with inheritance, Polymorphism-ad hoc polymorphism, pure polymorphism, method overriding, abstract classes, Object class, forms of inheritance- specialization, specification, construction, extension, limitation, combination, benefits of inheritance, costs of inheritance.

UNIT II

Packages

Defining a Package, CLASSPATH, 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): The Stream classes-Byte streams and Character streams, Reading console Input and Writing Console Output, File class, Reading and writing Files, Random access file operations, The Console class, Serialization, Enumerations, auto boxing, generics.

UNIT III

Exception Handling

Fundamentals of exception handling, Exception types, Termination or presumptive models, Uncaught exceptions, using try and catch, multiple catch clauses, nested try statements, throw, throws and finally, built- in exceptions, creating own exception sub classes.

Multithreading , Differences between thread-based multitasking and process-based multitasking, Java thread model, creating threads, thread priorities, creating multiple threads, synchronizing threads, inter thread communication, dead lock, Suspending, resuming, and stopping threads.

UNIT IV

The Collections Framework (java.util)

Collections overview, Collection Interfaces, The Collection classes- Array List, Linked List, Hash Set, Tree Set, Priority Queue, Array Deque. Accessing a Collection via an Iterator, Using an Iterator, The For-Each alternative, Map Interfaces and Classes, Comparators, Collection algorithms, Arrays, The Legacy Classes and Interfaces- Dictionary, Hash table ,Properties, Stack, Vector, More Utility classes, String Tokenizer, Bit Set, Date, Calendar, Random, Formatter.

UNIT V

GUI Programming

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, Inner classes, Anonymous Inner classes.

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 Object Oriented 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, and UniversitiesPress.
4. Programming in Java, S. Malhotra, S. Chudhary, 2nd edition, Oxford Univ.Press.

COMPUTER SCIENCE AND ENGINEERING

B.Tech IV Semester **L/T/P/C**
3/1/0/4

DESIGN AND ANALYSIS OF ALGORITHMS (D54PC5)

Course Objective:
Understand the design paradigms for developing an algorithm and analyzing it for a given problem.

Course Outcomes:
After learning the contents of this course, the student will be able to

1. Apply mathematical analysis methods to analyze the performance of algorithms and apply divide and conquer technique to solve the computing problems. L3
2. Demonstrate disjoint set operations and apply back tracking technique to solve the computing problems.L3
3. Apply Greedy method to solve various computing problems. – L3
4. Synthesize efficient algorithms in common engineering design situations using dynamic programming technique. L5
5. Solve complex problems using branch and bound technique and analyze NP hard and NP complete problems - L4

UNIT I
Introduction
Algorithm definition, Algorithm Specification, Performance Analysis-Space complexity, Time complexity, Randomized Algorithms. Divide and conquer- General method, applications – Binary search, Merge sort, Quicksort, Strassen's Matrix Multiplication.

UNIT II
Disjoint Set Operations
Disjoint set operations, union and find algorithms, AND/OR graphs, Connected Components and Spanning trees, Bi-connected components Back tracking-General method, applications the 8-queen problem, sum of subsets problem, graph coloring, Hamiltonian cycles.

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, Travelling salesperson 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 Book:

1. Fundamentals of Computer Algorithms, 2nd Edition, Ellis Horowitz, Sartaj Sahni and S.Raja sekharan, 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.Goodrichand 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 toAlgorithms,3rd Edition,T. H.Cormen,C.E.Leiserson,R.L. Rivest, and C.Stein, PHI

COMPUTER SCIENCE AND ENGINEERING

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

DATABASE MANAGEMENT SYSTEMS (D54PC6)

Course Objective:
Understand the fundamentals of relational systems including data models, databases.

Course Outcomes:
After learning the contents of this course, the student will be able to

1. Demonstrate the basic elements of a relational database management system and identify the data models for relevant problems. L3
2. Design entity relationship model and convert entity relationship diagrams into RDBMS L5
3. Formulate SQL queries for the given data base and apply the normalization techniques to the development of application software L5
4. Analyze the transaction management and concurrency control L4
5. Compare and contrast indexing and Hash based indexing L4

UNIT I
Introduction and Basic Concepts

File organization for conventional data management system, Higher-level file organization for DBMS, Database System Applications, Purpose of Database Systems, View of Data, Database Languages DDL, DML, Relational Databases, Database Design, Data Storage and Querying, Transaction Management, Database Architecture, Data Mining and Information Retrieval, Specialty Databases, Database Users and Administrators. Relational Model-Introduction to the Relational Model, Integrity Constraints over Relations, Enforcing Integrity, constraints, Querying relational data, Logical database Design: ER to Relational, Introduction to Views, Destroying /Altering Tables and Views.

UNIT II
Relational Algebra and Calculus

Preliminaries, Relational Algebra, Relational calculus Tuple relational Calculus, Domain relational calculus, Expressive Power of Algebra and calculus.
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, Conceptual Design for Large enterprises.

UNIT III
SQL

SQL data definition and Data types, Schema and catalog concepts in SQL, Queries, Constraints, Triggers: 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, No SQL database (MongoDB introduction).

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

Transactions, Transaction Concept, A Simple Transaction Model, Storage Structure, Transaction Atomicity and Durability, Transaction Isolation, serializability and Atomicity, Transaction Isolation Levels, Implementation of Isolation Levels. Concurrency Control, Lock–Based Protocols, Multiple Granularity, Time stamp-Based Protocols, Validation-Based Protocols, Multi-version Schemes. Recovery System-Failure Classification, Storage, Recovery and Atomicity, Recovery Algorithm, Buffer Management, Failure with loss of non-volatile storage, Early Lock Release and Logical Undo Operations, Remote Backup systems.

UNIT V
Indexing

Index Data Structures, and Comparison with File Organizations. Tree-Structured Indexing, Intuition for tree Indexes, Indexed Sequential Access Method (ISAM), B+ Trees: A Dynamic Index Structure, Search, Insert, Delete, Hash- Based Indexing, Static Hashing, Extendible hashing, Linear Hashing, Extendible vs. Linear Hashing.

Text Books:

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

Reference Books:

1. Database Systems, 6th edition, RElmasri, Shamkant B.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.
5. Introduction to Database Systems, C.J. Date, Pearson Education.

COMPUTER SCIENCE AND ENGINEERING

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

SOFTWARE ENGINEERING (D54PC7)

Course Objectives

1. The aim of the course is to provide an understanding of the working knowledge of the techniques for estimation, design, testing and quality management of large software development projects.
2. Topics include process models, software requirements, software design, software testing,
3. Software process/product metrics, risk management, quality management and UML diagrams

Course Outcomes

1. Ability to translate end-user requirements into system and software requirements, using e.g. UML, and structure the requirements in a Software Requirements Document (SRD).
2. Identify and apply appropriate software architectures and patterns to carry out high level design of a system and be able to critically compare alternative choices.
3. Will have experience and/or awareness of testing problems and will be able to develop a simple testing report

UNIT I

Introduction to Software Engineering: The evolving role of software, changing nature of software, software myths.

A Generic view of process: Software engineering- a layered technology, a process frame work, the capability maturity model integration (CMMI). Process models: The waterfall model, Incremental process models, Spiral model, proto typing and unified process.

Agile view of process: Agile process, agile process models.

UNIT II

Software Requirements: Functional and non-functional requirements, user requirements, system requirements, interface specification, the software requirements document.

Requirements engineering process: Feasibility studies, requirements elicitation and analysis, requirements validation, requirements management.

UNIT - III

Design Engineering: Design process and design quality, design concepts, the design model.

Creating an architectural design: software architecture, data design, architectural styles and patterns, architectural design.

Conceptual model of UML: SDLC, Basic structural modeling- class diagrams, Interaction diagrams- sequence diagrams, collaboration diagrams, use case diagrams, activity diagrams component diagrams, deployment diagrams.

UNIT IV

Testing Strategies: A strategic approach to software testing, test strategies for conventional software, black-box and white-box testing, validation testing, system testing, the art of debugging.

Metrics for Process and Products: Software measurement, metrics for software quality.

UNIT- V

Risk management: Reactive Vs proactive risk strategies, software risks, risk identification, risk projection, risk refinement, RMMM.

Quality Management: Quality concepts, software quality assurance, software reviews, formal technical reviews, statistical software quality assurance, software reliability, the ISO 9000 quality standards.

TEXT BOOKS:

1. Software Engineering, A practitioner's Approach- Roger S. Pressman, 6th edition, McGraw Hill International Edition.
2. Software Engineering- Sommerville, 7th edition, Pearson Education.
3. The unified modeling language user guide Grady Booch, James Rambaugh, Ivar Jacobson, Pearson Education.

REFERENCE BOOKS:

1. Software Engineering, an Engineering approach- James F. Peters, WitoldPedrycz, John Wiley.
2. Software Engineering principles and practice- Waman S Jawadekar, The McGraw-Hill Companies.
3. Fundamentals of object-oriented design using UML Meiler page-Jones: Pearson Education

COMPUTER SCIENCE AND ENGINEERING

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

WEB TECHNOLOGIES (D54PC8)

Course Objective:
To understand the concepts of PHP Language, processing of XML data with Java, server side programming with Java servlets and JSP.

Course Outcomes:
Upon completion of the course, the student will

1. Understand basics of server side scripting using PHP – L2
2. Illustrate well-formed XML programs and how to parse, use XML data with JAVA – L3
3. Design server side programming applications with servlets– L5
4. Develop programs using JSP for various applications –L5
5. Write programs with knowledge of client side scripting , validation of forms and AJAX programs – L4

UNIT I
Introduction to PHP: Declaring variables, data types, arrays, strings, operators, expressions, control structures, functions, Reading data from web form controls like text boxes, radio buttons, lists etc., Handling File Uploads, Connecting to database (MySQL as reference), executing simple queries, handling results, Handling sessions and cookies File Handling in PHP: File operations like opening, closing, reading, writing, appending, deleting etc. on text and binary files, listing directories.

UNIT II
XML: Introduction to XML, Defining XML tags, their attributes and values, Document Type Definition, XML Schemas, Document Object Model, XHTML Parsing XML Data – DOM and SAX Parsers in java.

UNIT III
Introduction to Servlets: Common Gateway Interface (CGI), Lifecycle of a Servlet, deploying a servlet, The Servlet API, Reading Servlet parameters, Reading
Initialization parameters, Handling Http Request & Responses, Using Cookies and Sessions, connecting to a database using JDBC.

UNIT IV
Introduction to JSP: The Anatomy of a JSP Page, JSP Processing, Declarations, Directives, Expressions, Code Snippets, implicit objects, Using Beans in JSP Pages, Using Cookies and session for session tracking, connecting to database in JSP.

UNIT V

Client-side Scripting: Introduction to Javascript: Javascript language – declaring variables, the scope of variables, functions, event handlers (click, onsubmit etc.), Document Object Model, Form validation. Simple AJAX application.

Text Books:

1. Web Technologies, Uttam K Roy, Oxford UniversityPress.
2. The Complete Reference PHP – Steven Holzner, TataMcGraw-Hill.

Reference Books:

1. Web Programming, building internet applications, Chris Bates 2nd edition, Wiley Dreamtech.
2. Java Server Pages –Hans Bergsten, SPD O'Reilly Java Script, D. Flanagan, O'Reilly, SPD.
3. Beginning Web Programming-Jon Duckett WROX.
4. Programming World Wide Web, R. W. Sebesta, Fourth Edition, Pearson.
5. Internet and World Wide Web – How to program, Dietel and Nieto, Pearson.

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

COMPUTER SCIENCE AND ENGINEERING

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

DATABASE MANAGEMENT SYSTEMS LAB (D54PC9)

Course Objective:

Emphasize on designing, developing and querying a database in the context of example database "Roadway travels".

Course Outcomes:

After learning the contents of this course, the student will be able to

1. Design and implement a database schema for a given problem. L5
2. Apply the normalization techniques for development of application software to realistic problems. L3
3. Formulate queries using SQL DML/DDL/DCL commands. L5
4. Develop application programs using PL/SQL. L5

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:

Reservation & 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 handled 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 queries 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.

Experiment1: 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.

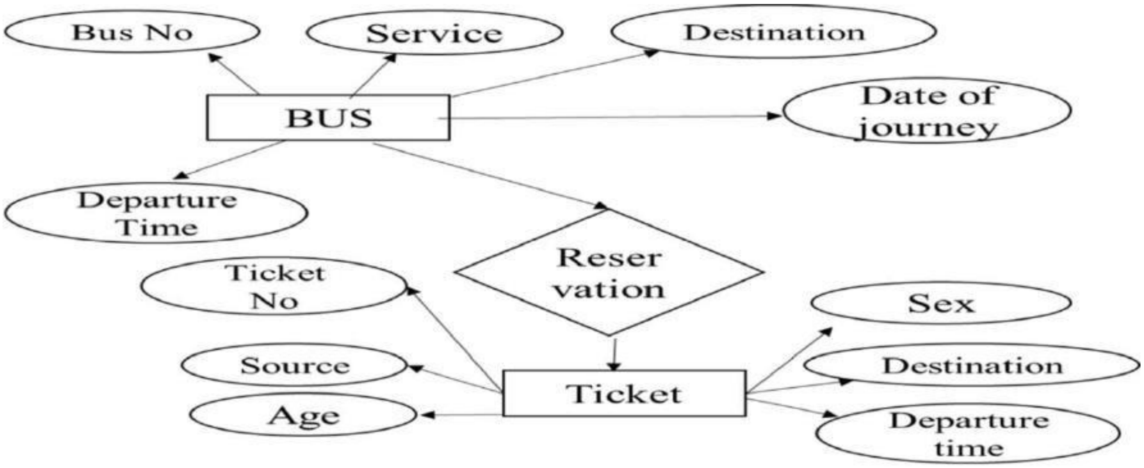
Primary Key Attributes: 1.TicketID (Ticket Entity) 2.PassportID (Passenger Entity)
3. Bus_No. (Bus Entity)

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

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



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.

Passenger Name	Age	Sex	Address	Passport ID	Ticket_ID

Experiment 4: Normalization

Database normalization is a technique for designing relational database tables to minimize duplication of information and, in so doing, to safe guard 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	Passport ID

Passport ID	Ticket_ID

You can do the second and third normal forms if required. Any, how Normalized tables are given at the end.

Experiment 5: Installation of Mysql and practicing DDL commands

Installation of MySql. In this week student will learn Creating databases, How to create tables, altering the database, dropping tables and databases if not required. Student 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.

Experiment 6: Practicing DML commands

DML commands are used to for managing data within schema objects. Some examples:

SELECT - retrieve data from the a database

INSERT - insert data into a table

UPDATE-updates existing data within a table

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 student 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 and45.
6. Display all the passengers names beginning with ‘A’
7. Display the sorted list of passengers names

Experiment 8: and Experiment 9: Querying (continued…)

Students are going to practice queries using Aggregate functions (COUNT, SUM, AVG, and MAX and MIN), GROUP BY, HAVING and Creation and dropping of Views.
Write a Query to display the Information present in the Passenger and cancellation tables.

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.
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 than1.Hint: Use GROUP BY, WHERE and HAVING CLAUSES.
5. Find the total number of cancelled seats.

Experiment 10: PL/SQL

1. Write a PL/SQL block for Addition of Two Numbers
2. Write a PL/SQL block for IF Condition
3. Write a PL/SQL block for IF and else condition
4. Write a PL/SQL block for greatest of three numbers using IFAND ELSEIF
5. Write a PL/SQL block for summation of odd numbers using for LOOP.

Experiment 11: 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;

Experiment 12: Procedures

Learn creation of stored procedure, Execution of procedure and modification of procedure. Practice procedures using the above database.

Eg: CREATE PROCEDURE myProc() BEGIN
SELECT COUNT (Tickets) FROM Ticket WHERE age>=40; End;

Experiment 13: Cursors

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.

CREATE PROCEDURE myProc(in_customer_id INT) BEGIN
DECLARE v_idINT;
DECLARE v_name VARCHAR(30);
DECLARE c1 CURSOR FOR SELECT stdId, stdFirstname FROM students WHERE stdId=in_customer_id;
OPEN c1;
FETCH c1 into v_id, v_name; Close c1; END;

Tables BUS
Bus No: Varchar: PK (public key) Source: Varchar Destination : Varchar Passenger
PPNO:Varchar(15)):PKName:Varchar(15)Age:int(4)Sex:Char(10):Male/Female
Address: VarChar (20)
Passenger_Tickets
PPNO: Varchar (15)): PK Ticket_No: Numeric (9) Reservation
PNR_No:Numeric(9):FKJourney_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
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
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

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

COMPUTER SCIENCE AND ENGINEERING

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

OBJECT ORIENTED PROGRAMMING THROUGH JAVA LAB (D54PC10)

Course Objective

Implement object oriented concepts using java programming in real time applications.

Course Outcomes

After completion of course, the student will be able to

1. Write programs for problems, using java collection frame work and abstract classesL3
2. Design and develop programs using objects and inheritance in Java language.L5
3. Write multithreaded programs.L3

List of Experiments to be performed during the Course

1. Use Eclipse or Net beans platform and acquaint with the various menus. Create a test project, add a test class, and run it. See how you can use auto suggestions, auto fill. Try code formatter and code refactoring like renaming variables, methods, and classes. Try debug step by step with a small program of about10 to 15 lines which contains at least one if else condition and a for loop.

2. Write a Java program that works as a simple calculator. Use a grid layout to arrange buttons for the digits and for the +, -,*, % operations. Add a text field to display the result. Handle any possible exceptions like divided by zero.

3. Write a Java program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a Number Format Exception. If Num2 were Zero, the program would throw an Arithmetic Exception. Display the exception in a message dialog box.

4. Write a Java program that implements a multi-thread application that has three threads. First thread generates random integer every 1 second and if the value is even, second thread computes the square of the number and prints. If the value is odd, the third thread will print the value of cube of the number.

5. Write a Java program for the following:
    a. Create a doubly linked list of elements.
    b. Delete a given element from the above list.
    c. Display the contents of the list after deletion.

6. Write a Java program that simulates a traffic light. The program lets the user select one of three lights: red, yellow, or green with radio buttons. On selecting a button, an appropriate message with “Stop” or “Ready” or “Go” should appear above the buttons in selected color. Initially, there is no message shown.

7. Write a Java program to create an abstract class named Shape that contains two integers and an empty method named print Area(). Provide three classes named Rectangle, Triangle, and Circle such that each one of the classes extends the class Shape. Each one of the classes contains only the method print Area() that prints the area of the given shape.

8. Suppose that a table named Table.txt is stored in a text file. The first line in the file is the header, and the remaining lines correspond to rows in the table. The elements are separated by commas. Write a java program to display the table using Labels in Grid Layout.

9. Write a Java program that handles all mouse events and shows the event name at the center of the window when a mouse event is fired (Use Adapter classes).

10. Write a Java program that loads names and phone numbers from a text file where the data is organized as one line per record and each field in a record are separated by a tab (\t). It takes a name or phone number as input and prints the corresponding other value from the hash table (hint: use hash tables).

11. Write a Java program that correctly implements the producer–consumer problem using the concept to finter thread communication.

12. Write a Java program to list all the files in a directory including the files present in all its Subdirectories.

13. Write a Java program that implements Quick sort algorithm for sorting a list of names in ascending order.

14. Write a Java program that implements Bubble sort algorithm for sorting in descending order and also show the number of interchanges occurred for the given set of integers.

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

COMPUTER SCIENCE AND ENGINEERING

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

WEB TECCNOLOGIES LAB (D54PC11)

Course Objective:
To enable the student to program web applications using the following technologies HTML, Javascript, AJAX, PHP, Tomcat Server, Servlets, JSP.

Course Outcomes:
After completion of this course, the student will be able to

1. Use LAMP Stack for web applications
2. Use Tomcat Server for Servlets and JSPs
3. Write simple applications with Technologies like HTML, Javascript, AJAX, PHP, Servlets, and JSPs, Parse XML files using Java (DOM and SAX parsers)
4. Connect to Database and get results.

List of Experiments:

1. Write an HTML page including JavaScript that takes a given set of integer numbers and shows them after sorting in descending order.

2. Write an HTML page including any required Javascript that takes a number from one text field in the range of 0 to 999 and shows it in another text field in words. If the number is out of range, it should show "out of range" and if it is not a number, it should show "not a number" message in the result box.

3. Write an HTML page that has one input, which can take multi-line text and a submit button. Once the user clicks the submit button, it should show the number of characters, words and lines in the text entered using an alert message. Words are separated with white space and lines are separated with new line character.

4. Write an HTML page that contains a selection box with a list of 5 countries. When the user selects a country, its capital should be printed next to the list. Add CSS to customize the properties of the font of the capital (color, bold and font size).

5. Create an XML document that contains 10 users information. Write a Java program, which takes User Id as input and returns the user details by taking the user information from the XML document using (a) DOM Parser and (b) SAX parser Implement the following web applications using (a) PHP, (b) Servlets and (c)JSP:

6. A user validation web application, where the user submits the login name and password to the server. The name and password are checked against the data already available in Database and if the data matches, a successful login page is returned. Otherwise a failure message is shown to the user.

7. Modify the 6th program to use an xml file instead of database.

8. Modify the 6th program to use AJAX to show the result on the same page below the submit button.

9. A simple calculator web application that takes two numbers and an operator (+, -, /, * and %) from an HTML page and returns the result page with the operation performed on the operands.

10. Modify the above program such that it stores each query in a database and checks the database first for the result. If the query is already available in the DB, it returns the value that was previously computed (from DB) or it computes the result and returns it after storing the new query and result in DB.

11. A web application takes a name as input and on submit it shows a hello page where is taken from the request. It shows the start time at the right top corner of the page and provides a logout button. On clicking this button, it should show a logout page with Thank You message with the duration of usage (hint: Use session to store name and time).

12. A web application that takes name and age from an HTML page. If the age is less than it should send a page with “Hello , you are not authorized to visit this site” message, where should be replaced with the entered name. Otherwise it should send “Welcome to this site” message.

13. A web application for implementation: The user is first served a login page which takes user’s name and password. After submitting the details the server checks these values against the data from a database and takes the following decisions. If name and password matches, serves a welcome page with user’s full name. If name matches and password doesn’t match, then serves “password mismatch” page If name is not found in the database, serves a registration page, where user’s full name is asked and on submitting the full name, it stores, the login name, password and full name in the database (hint: use session for storing the submitted login name and password)

14. A web application that lists all cookies stored in the browser on clicking “List Cookies” button. Add cookies if necessary.

COMPUTER SCIENCE AND ENGINEERING

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

CONSTITUTION OF INDIA (MC002)

Course Objectives: Students will be able to:

- Understand the premises informing the twin themes of liberty and freedom from a civil rightsperspective.
- To address the growth of Indian opinion regarding modern Indian intellectuals' constitutional role and entitlement to civil and economic rights as well as the emergence of nationhood in the early years of Indian nationalism.
- To address the role of socialism in India after the commencement of the Bolshevik Revolutionin 1917 and its impact on the initial drafting of the Indian Constitution.

Course Outcomes: Students will be able to:

- Discuss the growth of the demand for civil rights in India for the bulk of Indians before thearrival of Gandhi in Indian politics.
- Discuss the intellectual origins of the framework of argument that informed theconceptualization of social reforms leading to revolution in India.
- Discuss the circumstances surrounding the foundation of the Congress Socialist Party [CSP] under the leadership of Jawaharlal Nehru and the eventual failure of the proposal of direct elections through adult suffrage in the Indian Constitution
- Discuss the passage of the Hindu Code Bill of 1956.

Unit - 1 History of Making of the Indian Constitution- History of Drafting Committee.
Unit - 2 Philosophy of the Indian Constitution- Preamble Salient Features
Unit - 3 Contours of Constitutional Rights & Duties - Fundamental Rights

- Right to Equality
- Right to Freedom
- Right against Exploitation
- Right to Freedom of Religion
- Cultural and Educational Rights
- Right to Constitutional Remedies
- Directive Principles of State Policy
- Fundamental Duties.

Unit - 4 Organs of Governance: Parliament, Composition, Qualifications and Disqualifications, Powersand Functions, Executive, President, Governor, Council of Ministers, Judiciary, Appointment and Transfer of Judges, Qualifications, Powers and Functions

Unit - 5 Local Administration: District's Administration head: Role and Importance, Municipalities: Introduction, Mayor and role of Elected Representative, CEO of Municipal Corporation. Panchayat raj: Introduction, PRI: Zila Panchayat. Elected officials and their roles, CEO ZilaPanchayat: Position and role. Block level: Organizational Hierarchy (Different departments), Village level: Role of Elected and Appointed officials, Importance of grass root democracy

Unit - 6 Election Commission: Election Commission: Role and Functioning. Chief Election Commissioner and Election Commissioners. State Election Commission: Role and Functioning. Institute and Bodies for the welfare of SC/ST/OBC and women.

Suggested Reading:

1. The Constitution of India, 1950 (Bare Act), Government Publication.
2. Dr. S. N. Busi, Dr. B. R. Ambedkar framing of Indian Constitution, 1st Edition, 2015.
3. M. P. Jain, Indian Constitution Law, 7th Edn., Lexis Nexis, 2014.
4. D.D. Basu, Introduction to the Constitution of India, Lexis Nexis, 2015.