MARRI LAXMAN REDDY
INSTITUTE OF TECHNOLOGY AND MANAGEMENT
(AN AUTONOMOUS INSTITUTION)
(Approved by AICTE, New Delhi & Affiliated to JNTUH, Hyderabad)
Accredited by NBA and NAAC with 'A' Grade & Recognized Under Section2(f) & 12(B)of the UGC act,1956

B.Tech – INFORMATION TECHNOLOGY
II Year Course Structure And Syllabus (R20)
Applicable From 2020-21 Admitted Batch

II YEAR I SEMESTER

S. No.	Course Code	Course Title	Course Area	L	T	P	Credits	Internal (CIE)	External (SEE)	Total
1	2031203	Database Management Systems	PC	3	0	0	3	30	70	100
2	2030010	Business Economics and Financial Analysis	HSMC	3	0	0	3	30	70	100
3	2030004	Probability And Statistics	BS	3	0	0	3	30	70	100
4	2031204	Digital Logic Design	PC	3	1	0	4	30	70	100
5	2031205	Python Programming	PC	3	0	0	3	30	70	100
6	2031273	Database Management Systems Lab	PC	0	0	3	1.5	30	70	100
7	2031274	IT Workshop Lab	PC	0	0	3	1.5	30	70	100
8	2031275	Python Programming Lab	PC	0	0	3	1.5	30	70	100
9	2030022	Gender Sensitization	MC	2	0	0	0	-	-	-
Total Credits				17	1	9	20.5	240	560	800

II YEAR II SEMESTER

S. No.	Course Code	Course Title	Course Area	L	T	P	Credits	Internal (CIE)	External (SEE)	Total
1	2041206	Discrete Mathematics	PC	3	1	0	4	30	70	100
2	2040201	Basic Electrical Engineering	ES	3	0	0	3	30	70	100
3	2041207	Computer Organization &Microprocessors	PC	3	1	0	4	30	70	100
4	2041208	Design and Analysis of Algorithms	PC	3	0	0	3	30	70	100
5	2041209	JAVA Programming	PC	3	0	0	3	30	70	100
6	2040271	Basic Electrical Engineering Lab	ES	0	0	2	1	30	70	100
7	2041276	Design and Analysis of Algorithms through Java Lab	PC	0	0	3	1.5	30	70	100
8	2041277	Computer Organization &Microprocessors Lab using MASAM	PC	0	0	2	1	30	70	100
9	2040023	Constitution of India	MC	2	0	0	0	-	-	-
Total Credits				17	2	7	20.5	240	560	800

II - I

2031203: DATABASE MANAGEMENT SYSTEMS

B.Tech. II Year I -Sem

L T P C
3 0 0 3

Prerequisites:

A course on Data Structures and Linear Algebra.

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. Topics include data models, database design, relational model, relational algebra, transaction control, concurrency control, storage structures and access techniques.

Course Outcomes:
. **The students should be able to**
1. Gain knowledge of fundamentals of DBMS, database design and normal forms
2. Master the basics of SQL for retrieval and management of data.
3. Be acquainted with the basics of transaction processing and concurrency control.
4. Familiarity with database storage structures and access techniques

UNIT - I
Database Systems: A Historical Perspective, File Systems versus a DBMS**,** Relational Model, Levels of Abstraction in a DBMS, Data Independence, Structure of DBMS.

Introduction to Database Design: Database Design and ER Diagrams, Entities, Attributes, and Entity Sets, Relationships and Relationship Sets, Additional Features of the ER Model, Conceptual Design With ER Model.

UNIT - II
Relational Model: Introduction, Integrity constraints over relations, Enforcing integrity constraints, querying relational data, logical data base design, introduction to views, destroying/altering tables and views.

Relational Algebra and Calculus: Relational Algebra**,** Tuple relational Calculus, Domain relational calculus.

UNIT-III
SQL: Queries, Constraints, Triggers: Form of basic SQL query, UNION, INTERSECT, and EXCEPT, Nested Queries, aggregation operators, NULL values, complex integrity constraints in SQL, triggers and active data bases.

Schema refinement: Problems caused by redundancy, Decompositions, problems related to decomposition, Reasoning about Functional Dependencies, FIRST, SECOND, THIRD normal forms, BCNF, Lossless join decomposition, Multi-valued dependencies, FOURTH normal form, FIFTH normal form.

UNIT-IV

Transaction Management: ACID properties, Transactions and Schedules, Concurrent execution of transactions, Lock-based Concurrency control, Performance of locking, Transaction support in SQL, Introduction to crash recovery.

Concurrency control: Serializability and Recoverability, Introduction to lock management, Lock conversions, Dealing with dead locks, Spealized locking techniques, Concurrency control without locking.

UNIT – V

Storage and Indexing: Data on External Storage, File Organization and Indexing, Cluster Indexes, Primary and Secondary Indexes, Index data Structures, Hash Based Indexing, Tree base Indexing, Comparison of File Organizations, Indexes and Performance Tuning

Tree structured Indexing: Intuitions for tree Indexes, Indexed Sequential Access Methods (ISAM),

B+ Trees: A Dynamic Index Structure.

TEXT BOOKS:
1. Database Management Systems, Raghu Ramakrishanan, Johannes Gehrke, Tata McGraw Hill 3rd Edition
2. Database System Concepts, Silberschatz, Korth, McGraw hill, Vthedition.

REFERENCE BOOKS:
1. Database Systems design, Implementation, and Management, Peter Rob & Carlos Coronel, 7th Edition.
2. SQL The Complete Reference, James R. Groff, Paul N. Weinberg, 3rdEdition,
3. Oracle for Professionals, The X Team, S.Shah and V. Shah,SPD.
4. Database Systems Using Oracle: A Simplified guide to SQL andPL/SQL,Shah,PHI.

B.Tech. II Year I Sem. L T P C

3 0 0 3

Course Objective: To learn the basic Business types, impact of the Economy on Business and Firms specifically. To analyze the Business from the Financial Perspective.

Course Outcome: The students will understand the various Forms of Business and the impact of economic variables on the Business. The Demand, Supply, Production, Cost, Market Structure, Pricing aspects are learnt. The Students can study the firm's financial position by analyzing the Financial Statements of a Company.

UNIT – I Introduction to Business and Economics: Business: Structure of Business Firm, Types of Business Entities, Limited Liability Companies, Economics: Significance of Economics, Micro and Macro Economic Concepts, Business Cycle, Features and Phases of Business Cycle. Nature and Scope of Business Economics, Role of Business Economist.

Course outcomes:
By going through this unit, technical students can have the scope of learning about different economic concepts, business cycles and nature of business economists.

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, Demand Forecasting: Steps in Demand Forecasting, Methods of Demand Forecasting.

Course outcomes:
By going through this content, student can learn about different types of demand, its determinants and elasticity of demand concepts thoroughly and how to forecast the demand of different things by using different agreed upon techniques.

UNIT - III Production, Cost, Market Structures & Pricing: Production Analysis: Factors of Production, Production Function, Different Types of Production Functions. Cost analysis: Types of Costs, Short run and Long run Cost Functions. Market Structures: Features of Perfect competition, Monopoly, Oligopoly, and Monopolistic Competition. Pricing: Types of Pricing, Break Even Analysis, and Cost Volume Profit Analysis.

Course outcomes:
By reading this chapter, student can learn different pricing techniques in different market structures and different cost functions that determine products life cycle in a long term basis.

UNIT – IV Capital Budgeting: Importance of Capital Budgeting, methods of Capital Budgeting: Traditional Methods: Pay Back Period, Accounting Rate of Return, and Discounting Methods: Net Present Value, Profitability Index, Internal Rate of Return; Financial Analysis through Ratios: Concept of Ratio Analysis, Liquidity Ratios, Turnover

Ratios, Profitability Ratios, Proprietary Ratios, Solvency, Leverage Ratios (simple problems).

Course outcomes:
By going thoroughly through this unit, students can have the scope of learning about different techniques by which a project can be evaluated from financials perspective and utilization of ratios at different times to assess the business position for decision making.

UNIT - V 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.

Course outcomes:
Students can learn the methodology of accounting cycle which is valid from stakeholders' point of view and they can learn the comparison of the different firms at a time, so that they can take appropriate decision of either investment or to become an entrepreneur.

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. I.M. Pandey, Financial Management, 11th Edition, Kindle Edition, 2015.

REFERENCES:
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

2030004:PROBABILITY AND STATISTICS

L T P C
3 0 0 3

B.Tech. II Year I Semester-

Course Objectives: To learn
- The ideas of probability and random variables and various discrete and continuous probability distributions and their properties.
- The basic ideas of statistics including measures of central tendency.
- The statistical methods of studying data samples.
- The sampling theory and testing of hypothesis and making inferences.

Course Outcomes: After learning the contents of this paper the student must be able to
- Formulate and solve problems involving random variables and apply statistical methods for analysing experimental data.
- Apply discrete and continuous probability distributions.
- Classify the concepts of data science and its importance.
- Infer the statistical inferential methods based on small and large sampling tests.
- Interpret the association of characteristics through correlation and regression tools.

UNIT-I: Probability and Random Variables
Probability: Sample Space, Events, Probability of an Event, Additive Rules, conditional probability, independent events, Product Rule and Bayes' theorem.
Random variables: Discrete and continuous random variables. Expectation, Mean and Variance of random variables. Chebyshev's inequality.

UNIT-II: Probability Distributions & Estimation

Probability distribution-Binomial, Poisson approximation to the binomial distribution, uniform, exponential and Normal distribution. Estimation.: Estimating the Mean, Standard Error of a Point Estimate, Prediction Intervals, Tolerance Limits, Estimating the Variance, Estimating a Proportion for single mean, Difference between Two Means, between Two Proportions for Two Samples and Maximum Likelihood Estimation.

UNIT-III: Sampling theory and Small samples
Population and sample, parameters and statistics; sampling distribution of means (⊢ known)-central limit theorem, t-distribution, sampling distribution of means (⊢ unknown)-sampling distribution of variances-chi-square and F-distributions.

UNIT-IV: Testing of Hypothesis & Stochastic Process:
Testing of Hypothesis: Large sample test for single proportion, difference of proportions, single mean, difference of means.
Stochastic process: Introduction to Stochastic processes- Markov process. Transition Probability, Transition Probability Matrix, First order and Higher order Markov process, n- step transition probabilities, Markov chain, Steady state condition, Markov analysis.

UNIT-V: Curve Fitting for Statistical Data

Curve fitting by the method of least squares- fitting of straight lines, second degree parabolas and more general curves; Correlation and regression – Rank correlation.

TEXTBOOKS:
1. Ronald E. Walpole, Raymond H. Myers, Sharon L. Myers, keying Ye, Probability and statistics for engineers and scientists, 9th Edition, Pearson Publications.
2. Fundamentals of Mathematical Statistics, Khanna Publications, S C Guptas and V.K. Kapoor.

REFERENCES:
1. Miller and Freund's, Probability and Statistics for Engineers, 8th Edition, PearsonEducations
   S. Ross, A First Course in Probability, 6th Ed., Pearson Education India,2002

2031204:DIGITAL LOGIC DESIGN

(Common to CSE, IT, CSIT)

B.Tech. II Year I -Sem

L T   P C
3   1   0   4

Prerequisites: Nil

Course Objectives:
1. To understand basic number systems, codes and logical gates.
2. To understand the concepts of Boolean algebra.
3. To understand the use of minimization logic to solve the Boolean logic expressions..
4. To understand the design of combinational and sequential circuits.
5. To understand the state reduction methods for Sequential circuits.
6. To understand the basics of various types of memories.

Course Outcomes:
.   **The students should be able to**
1. Understand and explore the basics of computer networks and various protocols.
2. Understand number systems and codes.
3. Solve Boolean expressions using Minimization methods.
4. Design the sequential and combinational circuits.
5. State reduction methods to solve sequential circuits.
6. Understand and apply the memory and error detection and correction

UNIT - I
Digital Systems and Binary Numbers: Digital Systems, Binary Numbers, Number base conversions, Octal and Hexadecimal Numbers, complements, Signed binary numbers, Binary codes, Binary Storage and Registers, Binary logic.

Boolean Algebra And Logic Gates : Basic Definitions, Axiomatic definition of Boolean Algebra, Basic theorems and properties of Boolean algebra, Boolean functions canonical and standard forms, other logic operations, Digital logic gates, integrated circuits.

UNIT - II
Gate – Level Minimization : The map method, Four-variable map, Five-Variable map, product of sums simplification, Don't-care conditions, NAND and NOR implementation, other Two-level implementations, Exclusive – OR function.

UNIT-III
Combinational Logic : Combinational Circuits, Analysis procedure, Design procedure, Binary Adder-Subtractor, Decimal Adder, Binary multiplier, Magnitude comparator, Decoders, Encoders, Multiplexers, HDL for combinational circuits.

UNIT-IV

Synchronous Sequential Logic : Sequential circuits, latches, Flip-Flops, Analysis of clocked sequential circuits, HDL models for sequential circuits, State Reduction and Assignment, Design Procedure.

Registers and Counters: Registers, shift Registers, Ripple counters, synchronous counters, other counters, HDL for Registers and counters.

UNIT – V

Memory and Programmable Logic: Introduction, Random-Access Memory, Memory Decoding, Error Detection and correction Read-only memory, Programmable logic Array, programmable Array logic, Sequential Programmable Devices.

Asynchronous Sequential Logic : Introduction, Analysis Procedure, Circuits with Latches, Design Procedure, Reduction of state and Flow Tables, Race-Free state Assignment, Hazards, Design Example.

TEXT BOOKS:
   1. Digital Design – FourthEdition ,M.Morris Mano, Pearson Education/PHI.
   2. Fundamentals Of Logic Design, Roth, 5th Edition,Thomson.

REFERENCE BOOKS:
   1. Switching and Finite Automata Theory by Zvi. Kohavi, Tata McGraw Hill.
   2. Switching and Logic Design, C.V.S. Rao, Pearson Education
   3. Digital Principles and Design – Donald D.Givone, Tata McGraw Hill, Edition.
   4. Fundamentals of Digital Logic & Micro Computer Design , 5TH Edition, M. Rafiquzzaman John Wiley

2031205: PYTHON PROGRAMMING
(Common to All Branches)

B. Tech. II Year I -Sem
        L T P C
        3 0 0 3

  Prerequisites: Nil

Course Objectives:
1. Handle Strings and Files in Python.
2. Understand Lists, Dictionaries and Regular expressions in Python.
3. Understand FILES, Multithread programming in Python.
4. Understand GUI in Python.

Course Outcomes:
  .   **The students should be able to**
1. Examine Python syntax and semantics and be fluent in the use of Python flow control and functions.
2. Demonstrate proficiency in handling Strings and File Systems.
3. Create, run and manipulate Python Programs using core data structures like Lists, Dictionaries.
4. Develop programs using graphical user interface.

UNIT - I
Python Basics
Python Objects: Standard Types, Built-in Types, Internal Types, Standard Type Operators, Standard Type Built-in Functions, Categorizing the Standard Types, Unsupported Types.
Python Numbers: Introduction to Numbers, Integers, Floating Point Real Numbers, Complex Numbers, Operators, Built-in Functions.

UNIT - II
Conditionals and Loops-if, else, elif, for, while, break, continue, pass, List comprehensions, Generator expressions.
Sequences: Strings, Lists, and Tuples- Built-in Functions, Special features.
Mapping and Set Types: Dictionaries, Sets- Built-in Functions.

UNIT-III
Files and Input / Output: File Objects, File Built-in Functions, File Built-in Methods, File Built-in Attributes, Standard Files, Command-line Arguments, File System, File Execution, Persistent Storage Modules, Related Modules.
Exceptions: Exceptions in Python, Detecting and Handling Exceptions, Context Management, Exceptions as Strings, Raising Exceptions, Assertions, Standard Exceptions, Creating Exceptions, Exceptions and the sys Module.

UNIT-IV
Functions and Functional Programming –Calling Functions , Creating Functions, Passing Functions ,

Formal Arguments, Variable-Length Arguments, Functional Programming.
Modules–Modules and Files, Namespaces, Importing Modules, Module Built-in Functions, Packages, Related modules

UNIT – V
Multithreaded Programming: Introduction, Threads and Processes, Python Threads, the Global Interpreter Lock, Thread Module, Threading Module.
GUI Programming: Introduction, Tkinter and Python Programming, Brief Tour of Other GUIs, Related Modules and Other GUIs.

TEXT BOOKS:
1. Core Python Programming, Wesley J. Chun, Second Edition, Pearson.

REFERENCE BOOKS:
1. Think Python, Allen Downey, Green Tea Press
2. Introduction to Python, Kenneth A. Lambert, Cengage
3. Python Programming: A Modern Approach, VamsiKurama, Pearson
4. Learning Python, Mark Lutz, O'Really.

B.Tech. II Year I -Sem **L T P C**
                                          0 0 3 1.5

Prerequisites:

A course on Data Structures.

Course Objectives:

1. Introduce ER data model, database design and normalization
2. Learn SQL basics for data definition and data manipulation

Course Outcomes:

.   **The students should be able to**

1. Understand and explore the basics of computer networks and various protocols.
2. Design database schema for a given application and apply normalization
3. Acquire skills in using SQL commands for data definition and data manipulation.
4. Develop solutions for database applications using procedures, cursors and triggers.

Problem statement

Roadway Travels

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

- Reservation and Ticketing
- Cancellations

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 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 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 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 others 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. Passport ID (Passenger Entity)
3. Bus_No (Bus Entity)

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

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

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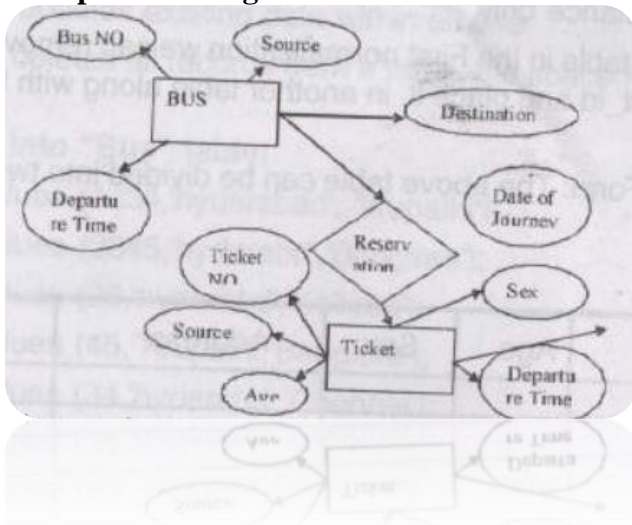
Note: The students is required to submit a document by drawing the E-R Diagram.

Experiment 3: Relational Model

Represent all the entities (Strong, Weak) in tabular fashion. Represent realtionships in a tabular fashion. There are different ways of representing relationships 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 E-R model. This is not a normalized table.

Passenger

Name	Age	Sex	Address	Ticket_id	Passport ID

Note: The students 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 multiple valued attribute Ticket_id and place it in another table along with the primary key of passenger.

First Normal Form: The above table can 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 practice DDL commands

Installation of MySQL. In this week you will learn Creating databases, How to create tables, altering the database, droping 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 commandsare used for managing data within schema objects. Some examples:

- SELECT - retrieve data from the 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

insert values into "Bus" table:

insert into Bus values (1234, 'hyderabad', 'tirupathi');

insert values into "Passenger" table:

insert into Passenger values(1, 45, 'ramesh', 45, 'M', 'abc123');
insert into Passenger values(56, 22, 'seetha', 32, 'F', 'abc55');

Few more Examples of DML commands:

SELECT * FROM Bus; (selects all the attributes and displays)
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 EXIST, 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 and Experiment 9: Querying (continued...)
You are going to practice queries using Aggregate functions (COUNT, SUM, AVG, MAX,and MIN), GROUP BY, HAVING and Creation and dropping of VIEWS.

1. Write a Query to display the information present in the Passenger and cancellation tables. Hint: Use UNION Operator.
2. Display the number of days in a week on which the 9W01 bus is available.
3. Find number of tickets booked for each PNR_No using GROUP BY CLAUSE. Hint: Use GROUP BY on PNR_No.
4. Find the distinct PNR numbers that are present.
5. 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.
6. Find the total number of cancelled seats.

Experiment 10: 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.
E.g:
CREATE TRIGGER updatecheck BEFORE UPDATE ON passenger FOR EACH ROW
BEGIN
  IF NEW.TickentNO> 60 THEN
    SET New.TickentNO = TicketNo;
  ELSE
    SET New.TicketNo = 0;
  END IF;
END

Experiment 11; Procedures
In this session you are going to learn Creation of stored procedure, Execution of procedure and modification of procedure. Practice procedures using the database.
E.g:
CREATE PROCEDURE myproc()
BEGIN
  SELECT   COUNT(Tickets)
  FROM   Ticket
  WHERE   age >= 40;
END;

Experiment 12: Cursors
In this week you need to do the following: 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_id INT;
    DECLARE v_nameVARCHAR(30);

    DECLARE c1 CURSOR FOR
    SELECT stdid, stdFirstname FROM studentsss WHERE stdid - in_customer_id;

    OPEN c1;
    FETCH c1 INTO v_id, v_name;
    CLOSE c1;
END;
```
Tables:
BUS
Bus No: VARCAHR : PK(primary key)
Source: VARCHAR
Destination: VARCHAR
Passenger
PPNO: VARCHAR(15) : PK
Name: 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) : FK
Journey_date: DATETIME(8)
No_of_seats: INT(8)
Address: VARCHRA(50)
Contact_No: NUMERIC(9) --> Should not less than 9 and Should not accept any other character other than interger
STATUS: CHAR(2) : Yes/No
Cancellation
PNR_No: NUMERIC(9) : FK
Journey_date: DATETIME(8)
No_of_seats: INT(8)
Address: VARCHRA(50)
Contact_No: NUMERIC(9) --> Should not less than 9 and Should not accept any other character other than interger
STATUS: CHAR(2) : Yes/No
Ticket
Ticket_No: NUMERIC(9) : FK
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.vanderLans, Pearson education.

2. Oracle PL/SQL, B.Rosenzweig and E.Silvestrova, Pearson education.
3. Oracle PL/SQL Programming, Steven Feuerstein, SPD.
4. SQL & PL/SQL for Oracle 10g, Black Book, Dr. P.S. Deshpande, Dream Tech.
5. Oracle Database 11g PL/SQL Programming, M. Mc Laughlin, TMH.
6. SQL Fundamentals, J.J. Patrick, Pearson Education.

2031274: IT WORKSHOP LAB
(Common to CSE,IT,CSIT)

B.Tech. II Year I -Sem

L T P C
0 0 3 1.5

Prerequisites: Nil

Course Objectives:
1. To train students on PC Hardware, Internet & World Wide Web and Productivity tools including Word, Excel, Power Point and Publisher.

Course Outcomes:
.   **The students should be able to**
1. Understand PC hardware.
2. Use tools MS-word and LATEX.

PC Hardware introduces the students to a personal computer and its basic peripherals, the process of assembling a personal computer, installation of system software like MS Windows, Linux and the required device drivers. In addition hardware and software level troubleshooting process, tips and tricks would be covered. **The students should work on working PC to disassemble and assemble to working condition and install Windows and Linux on the same PC. Students are suggested to work similar tasks in the Laptop scenario wherever possible. Internet & World Wide Web** module introduces the different ways   of hooking the PC on to the internet from home and workplace and effectively usage of the internet. Usage of web browsers, email, newsgroups and discussion forums would be covered. In addition, awareness of cyber hygiene, i.e., protecting the personal computer from getting infected with the viruses, worms and other cyber attacks would be introduced. **Productivity tools** module would enable the students in crafting professional word documents, excel spread sheets, power point presentations and personal web sites using the Microsoft suite of office tools and LaTeX. (**Recommended to use Microsoft office 2007 in place of   MS Office2003)**

PC Hardware
Task 1: Identify the peripherals of a computer, components in a CPU and its functions. Draw the block diagram of the CPU along with the configuration of each peripheral.

Task 2: Disassemble and assemble the PC back to working condition.

Task 3: Installation of MS windows on the personal computer or Laptop.

Task 4: Installation of Linux on the computer or Laptop. This computer should have windows installed. The system should be configured as dual boot with both windows and Linux.

Task 5: Hardware Troubleshooting: Sample PC which does not boot due to improper assembly or defective peripherals, identify the problem and fix it to get the computer back to working condition.

Task 6: Software Troubleshooting: Students should identify the problem and fix it to get the computer back to working condition if malfunction CPU is given.

Internet & World Wide Web

Task1: **Orientation & Connectivity Boot Camp: G**et connection of Local Area Network and access the Internet, website and email. Students have to simulate the WWW on LAN without internet connection.

Task 2: Web Browsers, Surfing the Web: Customize the web browsers with the LAN proxy settings, bookmarks, search toolbars and pop up blockers. Configure the plug-ins like Macromedia Flash and JRE for applets.

Task 3: **Search Engines & Netiquette:** Basic concept of search engines and how to use the search engines(GOOGLE, YAHOO etc)

Task 4: Cyber Hygiene: Installation of antivirus software and firewall.

LaTeX and Word

Task 1:Word Orientation: Accessing and overview of Toolbars, saving files, Using help and resources, rulers, format painter.

Task 2: To create sample certificate: Features to be covered - Formatting Fonts, Drop Cap, Applying Text effects, Character Spacing, Borders , Colors, Header , Footer, Date and Time. (Ex: Prize certificate, Participation certificate etc)

Task 3: Creating sample Bio-data: Features to be covered -Formatting Styles, Inserting table, Bullets and Numbering, Changing Text Direction, Cell alignment, Footnote, Hyperlink, Symbols, Spell Check, Track Changes.

Task 4 : Creating a Newsletter : Features to be covered - Table of Contents, Newspaper columns, Images from files and clipart, Drawing toolbar and Word Art, Formatting Images, Textboxes, Paragraphs.

Excel

Task 1:Excel Orientation: Accessing, overview of toolbars, saving excel files, Using help and resources.

Task 2: Creating a Scheduler - Features to be covered: Gridlines, Format Cells, Summation, auto fill, Formatting Text

Task 3 : Calculating GPA - .Features to be covered:- Cell Referencing, Formulae in excel – average, standard deviation, Charts, Renaming , Inserting worksheets, Hyper link, Count function, LOOKUP/VLOOKUP(Use Autonomous college CGPA)

Task 4: Performance Analysis - Features to be covered - Split cells, freeze panes, group, outline, Sorting, Boolean operators, logical operators, Conditional formatting.

Power Point(LaTeX/MS)

Task 1: Power point orientation: Features to be covered - PPT Orientation, Slide Layouts, Inserting Text, Word Art, Formatting Text, Bullets, Numbering, Auto Shapes, Lines and Arrows.

Task 2: Interactive presentation: Features to be covered-Hyperlinks, Inserting –Images, Clip Art, Audio, Video, Objects, Tables and Charts.

Task 3: Design: Features to be covered - Master Layouts**,** views, Insert, Background, Animations, Slide-show.

REFERENCE BOOKS:
1. Comdex Information Technology course tool kit, Vikas Gupta, WILEYDreamtech
2. The Complete Computer upgrade and repair book, 3rd Edition, CherylA Schmidt, WILEY Dreamtech
3. Introduction to Information Technology, ITL Education Solutions limited, PearsonEducation.
4. PC Hardware and A+Handbook, Kate J. Chase, PHI(Microsoft)
5. LaTeX Companion, Leslie Lamport,PHI/Pearson.
6. IT Essentials PC Hardware and Software Companion Guide Third Edition by David Anfinson and Ken Quamme. – CISCO Press, Pearson Education. IT Essentials PC Hardware and Software Labs and Study Guide Third Edition by Patrick Regan – CISCO Press , Pearson Education. Microsoft Office 2007: The Missing Manual - Chris Grover, Mathew MacDonald, E.A.Vander Veer O'reillyMedia

2031275: PYTHON PROGRAMMING LAB
(Common to all branches)

B.Tech. II Year I -Sem   **L   T   P   C**

                                                    0   0   3   1.5

 Prerequisites: Nil

Course Objectives:
1. Handle Strings and Files in Python.
2. Understand Lists, Dictionaries and Regular expressions in Python.
3. Understand FILES, Multithread programming in Python.
4. Understand GUI in python.

Course Outcomes:
.   **The students should be able to**
1. Examine Python syntax and semantics and be fluent in the use of Python flow control and functions.
2. Demonstrate proficiency in handling Strings and File Systems.
3. Create, run and manipulate Python Programs using core data structures like Lists, Dictionaries .
4. Develop programs using Graphical user interface.

Exercise 1 –Python Numbers
a) Write a program to determine whether a given year is a leap year, using the following formula: a leap year is one that is divisible by four, but not by one hundred, unless it is also divisible by four hundred. For example, 1992, 1996, and 2000 are leap years, but 1967 and 1900 are not. The next leap year falling on a century is 2400.
b) Write a program to determine the greatest common divisor and least common multiple of a pair of integers.
c) Create a calculator application. Write code that will take two numbers and an operator in the format: N1 OP N2, where N1 and N2 are floating point or integer values, and OP is one of the following: +, -, *, /, %, **, representing addition, subtraction, multiplication, division, modulus/remainder, and exponentiation, respectively, and displays the result of carrying out that operation on the input operands.
   Hint: You may use the string split() method, but you cannot use the exal () built-in function.

Exercise –2 Control Flow
a) Write a Program for checking whether the given number is a prime number or not.
b) Write a program to print Fibonacci series upto given n value.
c) Write a program to calculate factorial of given integer number.

Exercise 3 Control Flow -Continued
a) Write a program to calculate value of the following series 1+x-x2+x3-x4+----xn.
b) Write a program to print pascal triangle.

Exercise 4 – Python Sequences
a) Write a program to sort the numbers in ascending order and strings in reverse alphabetical order.
b) Given an integer value, return a string with the equivalent English text of each digit. For example,

an input of 89 results in "eight-nine" being returned. Write a program to implement it.

Exercise 5– Python Sequences
a) Write a program to create a function that will return another string similar to the input string, but with its case inverted. For example, input of "Mr. Ed" will result in "mR.eD" as the output string.
b) Write a program to take a string and append a backward copy of that string, making a palindrome.

Exercise 6– Python Dictionaries
a) Write a program to create a dictionary and display its keys alphabetically.
b) Write a program to take a dictionary as input and return one as output, but the values are now the keys and vice versa.

Exercise - 7 Files
a) Write a program to compare two text files. If they are different, give the line and column numbers in the files where the first difference occurs.
b) Write a program to compute the number of characters, words and lines in a file.

Exercise - 8 Functions
a) Write a function ball collide that takes two balls as parameters and computes if they are colliding. Your function should return a Boolean representing whether or not the balls are colliding.
b) Hint: Represent a ball on a plane as a tuple of (x, y, r), r being the radius
c) If (distance between two balls centers) <= (sum of their radii) then (they are colliding)
d) Find mean, median, mode for the given set of numbers in a list.
e) Write simple functions max2() and min2() that take two items and return the larger and smaller item, respectively. They should work on arbitrary Python objects. For example, max2(4, 8) and min2(4, 8) would each return 8 and 4, respectively.

Exercise - 9 Functions - Continued
a) Write a function nearlyequal to test whether two strings are nearly equal. Two strings a and b are nearly equal when a can be generated by a single mutation on b.
b) Write a function dups to find all duplicates in the list.
c) Write a function unique to find all the unique elements of a list.

Exercise - 10 - Functions - Problem Solving
a) Write a function cumulative_ product to compute cumulative product of a list of numbers.
b) Write a function reverse to reverse a list. Without using the reverse function.
c) Write function to compute GCD, LCM of two numbers. Each function shouldn't exceed one line.

Exercise - 11 GUI, Graphics
a) Write a GUI for an Expression Calculator usingtk
b) Write a program to implement the following figures using turtle
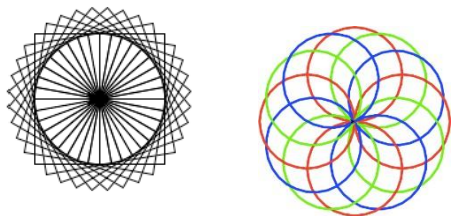
TEXT BOOKS:
1. Core Python Programming, Wesley J. Chun, Second Edition, Pearson.

REFERENCE BOOKS:

2. Think Python, Allen Downey, Green Tea Press
3. Introduction to Python, Kenneth A. Lambert, Cengage
4. Python Programming: A Modern Approach, VamsiKurama, Pearson
5. Learning Python, Mark Lutz, O'Really.

L/T/P/C
0/0/3/0

Course Objectives:
- To develop students' sensibility with regard to issues of gender in contemporary India.
- To provide a critical perspective on the socialization of men and women.
- To introduce students to information about some key biological aspects of genders.
- To expose the students to debates on the politics and economics of work.
- To help students reflect critically on gender violence.
- To expose students to more egalitarian interactions between men and women.

Course Outcomes:
- Students will have developed a better understanding of important issues related to gender in contemporary India.
- Students will be sensitized to basic dimensions of the biological, sociological, psychological and legal aspects of gender. This will be achieved through discussion of materials derived from research, facts, everyday life, literature and film.
- Students will attain a finer grasp of how gender discrimination works in our society and how to counter it.
- Students will acquire insight into the gendered division of labour and its relation to politics and economics.
- Men and women students and professionals will be better equipped to work and live together as equals.

UNIT – I
UNDERSTANDING GENDER
Gender: Why Should We Study It? (*Towards a World of Equals:* Unit -1)
Socialization: Making Women, Making Men (*Towards a World of Equals:* Unit -2)
Introduction. Preparing for Womanhood. Growing up Male. First lessons in Caste. Different Masculinities.

UNIT - II
GENDER AND BIOLOGY
Missing Women: Sex Selection and Its Consequences (*Towards a World of Equals:* Unit -4)
Declining Sex Ratio. Demographic Consequences.
Gender Spectrum: Beyond the Binary (*Towards a World of Equals:* Unit -10)
Two or Many? Struggles with Discrimination.

UNIT - III
GENDER AND LABOUR
Housework: the Invisible Labour (*Towards a World of Equals:* Unit -3)
‒My Mother doesn't Work.‖ ‒Share the Load.‖
Women's Work: Its Politics and Economics (*Towards a World of Equals:* Unit -7)
Fact and Fiction. Unrecognized and Unaccounted work. Additional Reading: Wages and Conditions of Work.

UNIT - IV
ISSUES OF VIOLENCE
Sexual Harassment: Say No! (*Towards a World of Equals:* Unit -6)
Sexual Harassment, not Eve-teasing- Coping with Everyday Harassment- Further Reading: ‒Chupulu‖.

Domestic Violence: Speaking Out (*Towards a World of Equals:* Unit -8)
Is Home a Safe Place? -When Women Unite [Film]. Rebuilding Lives. Additional
Reading: New Forums for Justice.
Thinking about Sexual Violence (*Towards a World of Equals:* Unit -11)
Blaming the Victim-‒I Fought for my Life….‖ - Additional Reading: The Caste Face
of Violence.

UNIT – V
GENDER: CO - EXISTENCE
Just Relationships: Being Together as Equals (*Towards a World of Equals:* Unit -12)
Mary Kom and Onler. Love and Acid just do not Mix. Love Letters. Mothers and
Fathers. Additional Reading: Rosa Parks-The Brave Heart.

TEXTBOOK
All the five Units in the Textbook, **"*Towards a World of Equals: A Bilingual Textbook
on Gender*"** written by A. Suneetha, Uma Bhrugubanda, Duggirala Vasanta, Rama
Melkote, Vasudha Nagaraj, Asma Rasheed, Gogu Shyamala, Deepa Sreenivas and Susie
Tharu and published by **Telugu Akademi, Hyderabad**,Telangana State in the year **2015**.

REFERENCE BOOKS:

1. Menon, Nivedita. Seeing like a Feminist. New Delhi: Zubaan-Penguin Books, 2012
2. Abdulali Sohaila. ‒I Fought For My Life…and Won.‖Available online
   at: http://www.thealternative.in/lifestyle/i-fought-for-my-lifeand-won-sohaila-abdulal/

II – II

B.Tech. II Year II Sem.

          L   T   P   C
          3   0   0   3

Prerequisites:

       A course on Data Structures and Mathematics.

Course Objectives:
1. To introduce the concepts of mathematical logic.
2. To introduce the concepts of sets, relations, and functions.
3. To perform the operations associated with sets, functions, and relations.
4. To introduce generating functions and recurrence relations.
5. To use Graph Theory for solving problems.

Course Outcomes:
.   **The students should be able to**
1. Apply mathematical logic to solve problems.
2. Understand sets, relations, functions, and discrete structures.
3. Use logical notation to define and reason about fundamental mathematical concepts such as sets, relations, and functions.
4. Formulate problems and solve recurrence relations.
5. Model and solve real-world problems using graphs and trees.

UNIT - I
Mathematical logic: Introduction, Statements and Notation, Connectives, well formed formula, Equivalence of formulas, Normal forms, Theory of inference for the statement calculus, predicate calculus, Inference theory of predicate calculu**s.**

UNIT - II
Set theory: Basic concepts of set theory, Set and Operations on sets, Relations and ordering, properties of binary relations in a set, Equivalence relation, Compatibility of relation, partial order relation, partial order set, Functions, Composition of functions, Inverse function, Recursive functions.

UNIT-III
Elementary Combinatorics: Basics of Counting, Combinations and Permutations, Enumeration of Combinations and permutations, Binomial Coefficients, Binomial and Multinational Theorems, Principle of Inclusion-Exclusion.

UNIT-IV
Recurrence Relations: Generating Functions of Sequences, Calculating Coefficients of generating functions, Recurrence relations, Solving recurrence relations by substitution and generating functions, Method of Characteristic roots, Solutions of Inhomogeneous Recurrence Relations.

UNIT – V

Graph Theory: Basic Concepts, Isomorphisms and Subgraphs, Trees and their Properties, Spanning Trees, Directed Trees, Binary Trees, Planar Graphs, Euler's Formula, Multigraphs and Euler Circuits, Hamiltonian Graphs, Chromatic Numbers, Four color problems.

TEXT BOOKS:
1. Discrete Mathematical Structures with Applications to Computer Science, J.P. Tremblay, R. Manohar, McGraw Hill education (India) Private Limited. (UNITS - I ,II)
2. Discrete Mathematics for Computer Scientists & Mathematicians, Joe L. Mott, Abraham Kandel, Theodore P. Baker, Pearson , 2nd ed. (Units - III, IV, V)

REFERENCE BOOKS:
1. Discrete Mathematics and its Applications, Kenneth H. Rosen, 7th Edition, McGraw Hill education (India) Private Limited.

2040201: BASIC ELECTRICAL ENGINEERING
(Common for ECE, CSE, CSC, CSD, CSM, CSIT & IT)

B.Tech. II Year II Sem.　　　　　　　　　　　　　　　　　　　　　　　**L　T/P　C**
　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　3　0/0　3

Course Prerequisites: Nil

Course Objectives:
- To analyse and solve electric circuits.
- To provide an understanding of basics in Electrical circuits.
- To identify the types of electrical machines for a given application.
- To explain the working principles of Electrical Machines and single phase transformers.

Course Outcomes

After completion of this course the student is able to
- Analyse Electrical circuits to compute and measure the parameters of Electrical Energy.
- Comprehend the working principles of Electrical DC Machines.
- Identify and test various electrical switchgear, single phase transformers and assess the ratings needed in given application.
- Comprehend the working principles of electrical AC machines.

UNIT-I DC Circuits:

Electrical circuit elements (R, L and C), voltage and current sources, Kirchhoff current and voltage laws, analysis of simple circuits with dc excitation. Superposition, Thevenin's and Norton's Theorems.

Unit-II AC Circuits:

Representation of sinusoidal waveforms, peak and rms values, phasor representation, real power, reactive power, apparent power and power factor. Analysis of single-phase ac circuits consisting of R, L, C, and RL, RC, RLC combinations (series only). Three phase balanced circuits, voltage and current relations in star and delta connections.

UNIT-III Transformers:

Ideal and practical transformer, equivalent circuit, losses in transformers, regulation and efficiency. Auto-transformer and three-phase transformer connections.

UNIT-IV: Electrical Machines:

Generation of rotating magnetic fields, Construction and working of a three-phase induction motor, Significance of torque-slip characteristic. Loss components and efficiency, starting and speed control of induction motor. Single-phase induction motor. Construction, working, torque-speed characteristic and speed control of separately excited dcmotor. Construction and working of synchronous generators.

UNIT-V Electrical Installations:

Components of LT Switchgear: Switch Fuse Unit (SFU), MCB, ELCB, MCCB, Types of Wires and Cables, Earthing. Types of Batteries, Important Characteristics for Batteries. Elementary calculations for energy consumption, power factor improvement and battery backup.

Text Books:
1. Basic Electrical Engineering - By M.S.Naidu and S. Kamakshaiah – TMH.
2. Basic Electrical Engineering –By T.K.Nagasarkar and M.S. Sukhija Oxford University Press.

Reference Books:
1. Theory and Problems of Basic Electrical Engineering by D.P.Kothari & I.J. Nagrath PHI.
2. Principles of Electrical Engineering by V.K Mehta, S.Chand Publications.
3. Essentials of Electrical and Computer Engineering by David V. Kerns, JR. J. David Irwin Pearson.

2041207:COMPUTER ORGANIZATION AND MICROPROCESSOR
(Common to CSE, IT, CSIT)

B.Tech. II Year II Sem.   **L   T   P   C**
                                                                      3   1   0   4

Prerequisites

A course on Digital Logic Design

Course Objectives:

1. To understand basic components of computers.
2. To understand the architecture of 8086 processor.
3. To understand the instruction sets, instruction formats and various addressing modes of 8086.
4. To understand the representation of data at the machine level and how computations are performed at machine level.
5. To understand the memory organization and I/O organization.
6. To understand the parallelism both in terms of single and multiple processors.

Course Outcomes:

The students should be able to

1. Understand the basic components and the design of CPU, ALU and Control Unit.
2. Understand memory hierarchy and its impact on computer cost/performance.
3. Understand the advantage of instruction level parallelism and pipelining for high performance Processor design.
4. Understand the instruction set, instruction formats and addressing modes of 8086.
5. Write assembly language programs to solve problems.

UNIT - I

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.

Micro Programmed Control: Control memory, Address sequencing, micro program example, design of control unit.

UNIT - II

8086 Architecture: 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.

8086 Instruction Set and Assembler Directive:

Machine language instruction formats, Addressing modes, Instruction set of 8086, Assembler directives and operators.

UNIT – III

Assembly Language Programming with 8086- Machine level programs, Machine coding the programs, Programming with an assembler, Assembly Language example programs.
Stack structure of 8086, Interrupts and Interrupt service routines, Interrupt cycle of 8086, Interrupt programming, Passing parameters to procedures, Macros, Timings and Delays.

UNIT - IV

Computer Arithmetic: Introduction, Addition and Subtraction, Multiplication Algorithms, Division Algorithms, Floating - point Arithmetic operations.

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

UNIT - V

Memory Organization: Memory Hierarchy, Main Memory, Auxiliary memory, Associate Memory, Cache Memory, Virtual memory.

Pipeline and Vector Processing: Parallel Processing, Pipelining, Arithmetic Pipeline, Instruction Pipeline, RISC Pipeline, Vector Processing, Array Processors.

TEXT BOOKS:
1. Computer System Architecture, M. Moris Mano, Third Edition, Pearson. (UNIST-I, IV, V)
2. Advanced Microprocessors and Peripherals, K M Bhurchandi, A.K Ray ,3rd edition, McGraw Hill India Education Private Ltd. (UNITS - II, III).

REFERENCE BOOKS:
1. Microprocessors and Interfacing, D V Hall, SSSP Rao, 3rd edition, McGraw Hill India Education Private Ltd.
2. Carl Hamacher, ZvonkoVranesic, SafwatZaky: 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, Elsevier, 2009.

B.Tech. II Year II Sem.	**L**	**T**	**P**	**C**
	3	**0**	**0**	**3**

Prerequisites:

Course on Programming for problem solving and Data Structures.

Course Objectives:
1. Introduces the notations for analysis of the performance of algorithms.
2. Introduces the data structure disjoint sets.
3. Describes major algorithmic techniques (divide-and-conquer, backtracking, dynamic programming, greedy, branch and bound methods) and mention problems for which each technique is appropriate.
4. Describes how to evaluate and compare different algorithms using worst-, average-, and best-case analysis.
5. Explains the difference between tractable and intractable problems, and introduces the problems that are P, NP and NP complete.

Course Outcomes:
.   **The students should be able to**
1. Analyze the performance of algorithms
2. Choose appropriate data structures and algorithm design methods for a specified application
3. Understand how the choice of data structures and the algorithm design methods impact the performance of programs.

UNIT - I
Introduction: Algorithm, Performance Analysis-Space complexity, Time complexity, Asymptotic Notations- Big oh notation, Omega notation, Theta notation and little ohnotation.
Disjoint Sets: Introduction, union and find Operations.
Divide and conquer: General method, applications-Binary search, Quick sort, Merge sort, Strassen's matrix multiplication

UNIT - II
Greedy method: General method, applications- knapsack problem, Job sequencing with deadlines, Minimum cost spanning trees, Single source shortest path problem.

UNIT-III
Dynamic Programming: General method, applications- All pairs shortest path problem, Optimal binary search trees, 0/1 knapsack problem, Reliability design ,Traveling sales person problem.

UNIT-IV
Backtracking: General method, applications, n-queen's problem, sum of subsets problem, graph

coloring, Hamiltonian cycles, knapsack problem.

UNIT – V

Branch and Bound: General method, applications - 0/1 knapsack problem, LC Branch and Bound solution, FIFO Branch and Bound solution, , Travelling 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, Ellis Horowitz, SatrajSahni and Rajasekharan, University Press.

REFERENCE BOOKS:
1. Design and Analysis of algorithms, Aho, Ullman and Hopcroft,Pearsoneducation.
2. Introduction to Algorithms, second edition,T.H.Cormen, C.E.Leiserson, R.L.Rivest, and C.Stein, PHI Pvt. Ltd./ Pearson Education.
3. Algorithm Design: Foundations, Analysis and Internet Examples, M.T. Goodrich and R. Tamassia, John Wiley and sons.

2041209: JAVA PROGRAMMING

(Common to All Branches)

B.Tech. II Year II Sem. **L T P C**

3 0 0 3

Prerequisites:

A course on Programming on problem solving

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:

. **The students should be able to**

1. Solve real world problems using OOP techniques.
2. Understand the use of abstract classes.
3. Solve problems using java collection framework and I/o classes.
4. Develop multithreaded applications with synchronization.
5. Develop applets for web applications.
6. Design GUI based applications

UNIT - I

Object oriented thinking: A way of viewing world – Agents, responsibility, messages, methods,

Classes and instances, class hierarchies – inheritance, method binding, overriding and exceptions

summary of oop concepts.

History of Java, Java buzzwords, data types, variables, scope and life time of variables, Type conversion and casting, arrays, operators, Operator Precedence, control statements.

Classes: Class fundamentals, Declaring Objects, methods, Constructors, this keyword, garbage collection, Overloading methods and constructors, Recursion.

UNIT - II

Inheritance, Packages and Interfaces – Inheritance basics, Using super, Creating a multilevel hierarchy, method overriding, Dynamic method dispatch, abstract classes, Using final with inheritance, Defining a package, Finding package and classpath, Access protection, importing packages, Defining an interface, implementing interface, applying interfaces, variables in interface and extending interfaces.

UNIT-III
Exception handling and Multithreading- Exception types, uncaught exceptions, using try and catch, Multiple catch classes, nested try statements, throw, throws and finally. Java's built-in exceptions, chained exceptions, creating own exception sub classes. Java thread model, thread priorities, synchronization, messaging, thread class and runnable interface, creating thread, creating multiple threads, thread priorities, synchronizing threads, interthread communication, thread life cycle.

UNIT-IV
Event Handling : Events, Event sources, Event Listeners, Event classes, Event listener interface, Handling mouse and keyboard events, Adapter classes, The AWT class hierarchy, AWT controls- labels, buttons, scrollbars, text field, check box, check box groups, choices, handling lists, dialogs, Menubar, layout manager – layout manager types – border, grid, flow, card and grid bag.

UNIT – V
Applets – Types, Applet basics, Applet architecture, applet skeleton, simple applet display methods, passing parameters to applets. Swing – Introduction, MVC connection, components, containers, exploring swing- Japplet, Jframe and Jcomponent, Icons and Labels, text fields, buttons – The Jbutton class, Check boxes, Radio buttons, Combo boxes, Tabbed Panes, Scroll Panes, Trees, and Tables.

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.(UNIT-I first part)

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, University 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.

2040271: BASIC ELECTRICAL ENGINEERING LAB

A.Tech. II Year II Sem.   L T P C
                                                                                                                          0 0 2 1

Course Objectives:

To analyze a given network by applying various electrical laws and network theorems

- To know the response of electrical circuits for different excitations
- To calculate, measure and know the relation between basic electrical parameters.
- To analyze the performance characteristics of DC and AC electrical machines

Course Outcomes:

- Get an exposure to basic electrical laws.
- Understand the response of different types of electrical circuits to different excitations.
- Understand the measurement, calculation and relation between the basic electrical parameters
- Understand the basic characteristics of transformers and electrical machines.

List of experiments/demonstrations:

1. Verification of Ohms Law
2. Verification of KVL and KCL
3. Verification of superposition theorem.
4. Verification of Thevenin's and Norton's theorem.
5. Resonance in series RLCcircuit.
6. Calculations and Verification of Impedance and Current of RL, RC and RLCseries circuits.
7. Measurement of Voltage, Current and Real Power in primary and Secondary Circuits of a Single Phase Transformer.
8. Load Test on Single Phase Transformer (Calculate Efficiency and Regulation)
9. Three Phase Transformer: Verification of Relationship between Voltagesand Currents (Star-Delta, Delta-Delta, Delta-star, Star-Star)
10. Measurement of Active and Reactive Power in a balanced Three-phase circuit.
11. Performance Characteristics of a Separately/Self Excited DC Shunt/Compound Motor.
12. Torque-Speed Characteristics of a Separately/Self Excited DCShunt/Compound Motor.
13. Performance Characteristics of a Three-phase Induction Motor.
14. Torque-Speed Characteristics of a Three-phase Induction Motor.
15. No-Load Characteristics of a Three-phase Alternator.

2041276: DESIGN AND ANALYSIS OF ALGORITHMS
LAB USING JAVA
(Common to CSE,IT,CSIT)

B.Tech. II Year II Sem.   L   T   P   C
                                                               0   0   3   1.5

Prerequisites:
Programming for problem solving lab

Course Objectives:
1. It covers various concepts of java programming language
2. It introduces the feasible and optimal solutions by using the different design methods

Course Outcomes:
.   **The students should be able to**
1. Develop the feasible and optimal solutions using Greedy and dynamic programming.
2. Develop the feasible and optimal solutions using Backtracking and Dynamic programming

List of Programs:
1.
   a. Use Eclipse or Net bean 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 about 10 to 15 lines which contains at least one if else condition and a for loop.
   b. Write a java program that prints all real solutions to the quadratic equation $ax^2 +bx+c=0$. Read in a, b, c and use the quadratic formula.
   c. Write a java program to implement Fibonacci series.
2.
   a. Write a java program to implement method overloading and constructors overloading.
   b. Write a java program to implement method overriding.
   c. 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.
3.
   a. Write a java program to check whether a given string is palindrome.
   b. 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 namedRectangle, Triangle, and Circle such that each one of the classes extends the classShape. Each one of the classes contains only the method print Area () that prints thearea of the given shape.
4. Write a program to implement Knapsack problem using greedy method.
5. Write a program to implement Prim's minimum cost spanning tree using Greedy Method
6. Write a program to implement Kruskal's minimum cost spanning tree using Greedy Method
7. Write a program to implement Job sequencing with deadlines using Greedy Method
8. Write a program to implement Single source shortest path problem using Greedy Method
9. Write a program to implement All pairs Shortest path using Dynamic Programming

10. Write a program to implement Optimal Binary Search Tree using Dynamic Programming
11. Write a program to implement 0/1 Knapsack problem using Dynamic Programming
12. Write a program to implement n-Queen's problem using backtracking method.
13. Write a program to implement Sum of subsets using backtracking method.
14. Write a program to implement Graph Coloring using backtracking method.
15. Write a program to implement Travelling sales person using branch and bound, dynamic programming

TEXT BOOKS:
1. Fundamentals of Computer Algorithms, Ellis Horowitz, SatrajSahni and Rajasekharan, University Press.

REFERENCE BOOKS:
1. Design and Analysis of algorithms, Aho, Ullman and Hopcroft,Pearson education.
2. Introduction to Algorithms, second edition,T.H.Cormen, C.E.Leiserson, R.L.Rivest, and C.Stein, PHI Pvt. Ltd./ PearsonEducation.
3. Algorithm Design: Foundations, Analysis and Internet Examples, M.T. Goodrich and R. Tamassia, John Wiley andsons
4. Java The Complete Reference, Herbert Schildt's, 9th Edition, TATA McGRAW –HILL.

2041277:COMPUTER ORGANIZATION AND MICROPROCESSORS USING MASAM
(Common to CSE, IT, CSIT)

B.Tech. II Year II Sem.

L	T	P	C
0	0	2	1

Prerequisites: Nil

Course Objectives:
1. Implement assembly language programs in MASM software.

Course Outcomes:
.   **The students should be able to**
1. Understand and apply the MASM software
2. Implement assembly language programs

List of Experiments
1.   Write assembly language programs to evaluate the expressions:
    i) a = b + c −d * e ii) z = x * y + w − v +u / k
    a. Considering 8-bit, 16 bit and 32-bit binary numbers as b, c, d, e.
    b. Considering 2-digit, 4 digit and 8-digit BCD numbers.
    Take the input in consecutive memory locations and results also Display the results by using "int xx" of 8086. Validate program for the boundary conditions.
2.   Write an ALP of 8086 to take N numbers as input and do the following operations on them. a. Arrange in ascending and descending order.
3.   Find maximum and minimum a. Find average Considering 8-bit, 16-bit binary numbers and 2-digit, 4 digit and 8-digit BCD numbers. Display the results by using "int xx" of 8086. Validate program for the boundary conditions.
4.   Write an ALP program to print the Fibonacci series.
5.   Write an ALP Program to find even or odd number using macros.
6.   Write a simple program in ALP using procedures with arguments.
7.   Write an ALP program to find prime no in a list.
8.   Write an ALP of 8086 to take a string of as input (in 'C' format) and do the following Operations on it. a. Find the length b. Find it is Palindrome or not
9.   Write an ALP of 8086 to do following operations.
    a)   find whether given string substring or not.
    b)   Reverse of a string
    c)   Concatenate by taking another sting Display the results by using "int xx" of 8086.
10.   Write the ALP to implement the above operations as procedures and call from the main procedure.
11.   Write an ALP of 8086 to find the factorial of a given number as a Procedure and call from the main program which display the result.
TEXT BOOKS:
1.   Switching theory and logic design −A. Anand Kumar PHI, 2013

2. Advanced microprocessor & Pieperar-A. K. Ray and K. M. Bherchandavi, TMH, 2nd edition.

REFERENCE BOOKS:
1. Switching and Finite Automatic theory-ZviKohavi, NirajK.JhaCambridge ,3rd edition
2. Digital Design –Morris Mano, PHI, 3rd edition
3. Microprocessor and Interfacing –Douglas V. Hall, TMGH 2nd edition.

2040023: CONSTITUTION OF INDIA

The Constitution of India is the supreme law of India. Parliament of India cannot make any law which violates the Fundamental Rights enumerated under the Part III of the Constitution. The Parliament of India has been empowered to amend the Constitution under Article 368, however, it cannot use this power to change the ―basic structure‖ of the constitution, which has been ruled and explained by the Supreme Court of India in its historical judgments. The Constitution of India reflects the idea of Constitutionalism‖ – a modern and progressive concept historically developed by the thinkers of ‒liberalism‖ – an ideology which has been recognized as one of the most popular politicalideology and result of historical struggles against arbitrary use of sovereign power by state. The historic revolutions in France, England, America and particularly European Renaissance andReformation movement have resulted into progressive legal reforms in the form of ‒constitutionalism‖in many countries. The Constitution of India was made by borrowing models and principles from many countries including United Kingdom and America.

The Constitution of India is not only a legal document but it also reflects social, political and economic perspectives of the Indian Society. It reflects India's legacy of ‒diversity‖. It has been said that Indian constitution reflects ideals of its freedom movement; however, few critics have argued that it does not truly incorporate our own ancient legal heritage and cultural values. No law can be ‒static‖and therefore the Constitution of India has also been amended more than one hundred times. These amendments reflect political, social and economic developments since the year 1950. The Indian judiciary and particularly the Supreme Court of India has played an historic role as the guardian of people. It has been protecting not only basic ideals of the Constitution but also strengthened the same through progressive interpretations of the text of the Constitution. The judicial activism of the Supreme Court of India and its historic contributions has been recognized throughout the world and it gradually made it ‒as one of the strongest court in the world‖.

Course content
1. Meaning of the constitution law and constitutionalism
2. Historical perspective of the Constitution of India
3. Salient features and characteristics of the Constitution of India
4. Scheme of the fundamental rights
5. The scheme of the Fundamental Duties and its legal status
6. The Directive Principles of State Policy – Its importance and implementation
7. Federal structure and distribution of legislative and financial powers between the Union and the States
8. Parliamentary Form of Government in India – The constitution powers and status of the President of India
9. Amendment of the Constitutional Powers and Procedure
10. The historical perspectives of the constitutional amendments in India
11. Emergency Provisions: National Emergency, President Rule, Financial Emergency
12. Local Self Government – Constitutional Scheme in India
13. Scheme of the Fundamental Right to Equality
14. Scheme of the Fundamental Right to certain Freedom under Article 19
15. Scope of the Right to Life and Personal Liberty under Article 21