# 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 – Computer Science and Engineering (Data Science)**
**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 | Hours Per Week | | | Credits | Scheme of Examination Maximum Marks | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | | L | T | P | | Internal (CIE) | External (SEE) | Total |
| 1 | 2036701 | Database Management Systems | PC | 3 | 0 | 0 | 3 | 30 | 70 | 100 |
| 2 | 2036702 | Operating Systems | PC | 3 | 0 | 0 | 3 | 30 | 70 | 100 |
| 3 | 2036703 | Digital Logic Design & computer Organization | PC | 3 | 1 | 0 | 4 | 30 | 70 | 100 |
| 4 | 2030004 | Probability & Statistics | BS | 3 | 0 | 0 | 3 | 30 | 70 | 100 |
| 5 | 2036704 | Python Programming | PC | 3 | 0 | 0 | 3 | 30 | 70 | 100 |
| 6 | 2036771 | Operating Systems Lab | PC | 0 | 0 | 3 | 1.5 | 30 | 70 | 100 |
| 7 | 2036772 | Database Management Systems Lab | PC | 0 | 0 | 3 | 1.5 | 30 | 70 | 100 |
| 8 | 2036773 | Python Programming Lab | PC | 0 | 0 | 3 | 1.5 | 30 | 70 | 100 |
| 9 | 2030025 | 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 | Hours Per Week | | | Credits | Scheme of Examination Maximum Marks | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | | L | T | P | | Internal (CIE) | External (SEE) | Total |
| 1 | 2046705 | Discrete Mathematics | PC | 3 | 1 | 0 | 4 | 30 | 70 | 100 |
| 2 | 2040201 | Basic Electrical Engineering | ES | 3 | 0 | 0 | 3 | 30 | 70 | 100 |
| 3 | 2040010 | Business Economics and Financial Analysis | HSMC | 3 | 0 | 0 | 3 | 30 | 70 | 100 |
| 4 | 2046706 | JAVA Programming | PC | 3 | 0 | 0 | 3 | 30 | 70 | 100 |
| 5 | 2046707 | Introduction to Data Science | PC | 3 | 0 | 0 | 3 | 30 | 70 | 100 |
| 6 | 2046708 | Data Warehousing & Data Mining | PC | 2 | 0 | 0 | 2 | 30 | 70 | 100 |
| 7 | 2040271 | Basic Electrical Engineering Lab | ES | 0 | 0 | 2 | 1 | 30 | 70 | 100 |
| 8 | 2046774 | Data Warehousing & Data Mining Lab | PC | 0 | 0 | 2 | 1 | 30 | 70 | 100 |
| 9 | 2046775 | JAVA Programming Lab | PC | 0 | 0 | 3 | 1.5 | 30 | 70 | 100 |
| 10 | 2040023 | Constitution of India | MC | 2 | 0 | 0 | 0 | - | - | - |
| | | Total Credits | | 19 | 1 | 7 | 21.5 | 270 | 630 | 900 |

# II - I

**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 II Year - Syllabus**

**2036701-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.

## 2036702-OPERATING SYSTEMS

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

**Prerequisites**

Nil

## Course Objectives

1. Provide an introduction to operating system concepts (i.e., processes, threads, scheduling, synchronization, deadlocks, memory management, file and I/O subsystems and protection).
2. Introduce the issues to be considered in the design and development of operating system
3. Introduce basic Unix commands, system call interface for process management, interprocess communication and I/O in Unix.

## Course Outcomes
### The Students should be able to
1. Control access to a computer and the files that may be shared
2. Demonstrate the knowledge of the components of computer and their respective roles in computing.
3. Ability to recognize and resolve user problems with standard operating environments.
4. Gain practical knowledge of how programming languages, operating systems, and architectures interact and how to use each effectively.

## UNIT - I
**Operating System Introduction**: What is an operating system do, computer system organization, computer system architecture, operating system structure - operating system operations, process management, memory management, operating system services, system calls, types of system calls.

## UNIT - II
**Process**:-Process concepts, process scheduling, Operations on processes, Interposes Communication, multithreading models, thread libraries.
**Process Scheduling**: -Scheduling Criteria, scheduling algorithms, thread scheduling Multiple - Processor Scheduling.

## UNIT - III

**Sychronization:-**background, the critical section problem, peters's solution, synchronization hardware, semaphores classical problems of synchronization, monitors.
**Deadlocks** - System Model, Deadlocks Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, and Recovery from Deadlock.

## UNIT - IV
**Memory Management and Virtual Memory** – background, swapping, contiguous memory allocation, paging structure of the page table, segmentation.
Virtual memory-background, demant paging page replacement allocation of frames thrashing.

## UNIT – V

**File Systems**

File system and implementing file system, file concept access methods, directory and disk structure, file system mounting file sharing, protection file system structure, file system implementation, directory implementation, allocation methods, free space management efficiency and performance, recovery ,NFS.

.

**Text Books**
1. Operating System concepts- Abraham Silberchatz, Peter B. Galvin, Greg Gagne 8$^{th}$ Edition, John Wiley
2. Advanced programming in the Unix environment, W.R.Stevens, Pearson education.

**References**
1. Operating Systems – Internals and Design Principles, Stallings, 5$^{th}$ Edition, Pearson Education/PHI, 2005.
2. Operating System A Design Approach-Crowley, TMH.
3. Modern Operating Systems, Andrew S Tanenbaum 2$^{nd}$ edition, Pearson/PHI
4. Unix programming environment, Kernighan and Pike, PHI. / Pearson Education
5. Unix Internals The New Frontiers, U.Vahalia, Pearson Education.

## 2036703-DIGITAL LOGIC DESIGN AND COMPUTER ORGANIZATION

**B.Tech. II Year I-Sem**

**L T P C**

**3 1 0**

**4**

**Prerequisites**

Nil

**Course Objectives:**

1. To understand the basic theoretical concepts of digital systems like the binary system and Boolean algebra.
2. To express real life problem in logic design terminology.
3. To use Boolean algebraic formulations to design digital systems. To design using combinational/sequential circuits
4. To understand the concepts of memory units
5. To explain the functions of the various I/O .

**Course Outcomes**
**The students should be able to**
1. Understand logic gates and binary codes.
2. Design the sequential and combinational circuits.
3. Understand computer arithmetic and addressing modes
4. Understand and apply the concepts of memory units and I/O

**UNIT- I**
**Basic Structure of Computers:** Basic operational concepts, Bus structures, Software, Performance, multiprocessors and multi computers, Historical perspective.
**Digital Systems and Binary Numbers**:- Complements, Signed binary numbers, Binary codes, Binary storage and registers, binary logic.
**Boolean algebra and logic gates,** gate level minimization basic definitions axiomatic of Boolean algebra, basic thermos, canonical and standard forms, the map methods four variable map, five variable map.

**UNIT- II**
**Combinational logic**: - combination circuits, binary adder substracter decimal adder binary multiplier, Magnetic compotator, decoders, encoders, multiplexers.
**Synchronous Sequential Logic:** Sequential circuits, latches, Flip-Flops, Analysis of clocked sequential circuits, State Reduction and Assignment, Design Procedure.
**Registers and Counters:** Registers, shift Registers, Ripple counters, synchronous counters.

**UNIT- III**
**Arithmetic:** Addition and subtraction of signed numbers, Design of fast adder, Multiplication of positive Numbers, signed operand multiplication, fast multiplication, integer division, floating point numbers.
**Machine Instruction and Programs:** Memory Locations and Addresses, Memory operations, instructions and instruction sequencing, Addressing Modes

**UNIT- IV**

**Basic Processing Unit:** Multiple Bus Organization, Hardwired Control, Microprogrammed Control **Memory Systems**: Concept of Memory, RAM, ROM memories, cache memories, virtual memory, secondary storage, memory management requirements.

**UNIT- IV**

**Input / Output Organization:** Introduction to I/O, Interrupts- Hardware, Enabling and disabling Interrupts, Device Control, Direct memory access, buses, interface circuits, standard I/O Interfaces.

**TEXT BOOKS:**

1. Digital Design – Fourth Edition , M.Morris Mano, Pearson Education/PHI
2. Computer Organization – Carl Hamacher, ZvonkoVranesic, SafwatZaky, fifth edition, McGraw Hill.

**REFERENCE BOOKS:**

1. Computer Architecture and Organization- An Integrated Approach, Miles Murdocca, Vincent Heuring, Second Edition, Wiley India.
2. Computer Systems Architecture – M.Moris Mano, IIIrd Edition, Pearson.
3. Computer Organization and Architecture – William Stallings Sixth Edition, Pearson
4. Computer- organization and Design- David A. Paterson and John L.Hennessy-Elsevier.
5. Fundamentals or Computer Organization and Design, - SivaramaDandamudi Springer Int. Edition.
6. Digital Design – Third Edition, M.Morris Mano, Pearson Education/PHI.
7. Fundamentals of Logic Design, Roth, 5th Edition, Thomson.

## 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, conditionalprobability, independent events, Product Rule and Bayes' theorem.
**Random variables:** Discrete and continuous random variables. Expectation, Mean and Varianceof 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, StandardError 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 ($\sigma$ known)-central limit theorem, t-distribution, sampling distribution of means ($\sigma$ 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, singlemean, 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 andmore general curves; Correlation and regression – Rank correlation.

**TEXTBOOKS:**

1. Ronald E. Walpole, Raymond H. Myers, Sharon L. Myers, keying Ye, Probability andstatistics 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, 8thEdition,PearsonEducations

   S. Ross, A First Course in Probability, 6th Ed., Pearson Education India,2002

# 2036704-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.

# 2036771-OPERATING SYSTEMS LAB (Using UNIX/LINUX)

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

                                                                 **0  0  3 1.5**

## Course Objectives

1. To provide an understanding of the design aspects of operating system concepts through simulation.
2. Introduce basic Unix commands, system call interface for process management, interprocess communication and I/O in Unix

## Course Outcomes

### The students should be able to

1. Simulate and implement operating system concepts such as scheduling, deadlock management, file management and memory management.
2. Able to implement C programs using Unix system calls

## List of Experiments

1. Write C programs to simulate the following CPU Scheduling algorithms
     a) FCFS        b)SJF        c)Round Robin       d) priority
2. Write programs using the I/O system calls of UNIX/LINUX operating system (open, read, write, close, fcntl, seek, stat, opendir, readdir)
3. Write a C program to simulate Bankers Algorithm for Deadlock Avoidance and Prevention.
4. Write a C program to implement the Producer – Consumer problem using semaphores using UNIX/LINUX system calls.
5. Write C programs to illustrate the following IPC mechanisms
     a) Pipes  b)FIFOs      c)Message Queues   d) Shared Memory
6. Write C programs to simulate the following memory management techniques
     a) Paging b)Segmentation

## Text Books

1.Operating System Principles- Abraham Silberchatz, Peter B. Galvin, Greg Gagne 7th Edition, JohnWiley

2. Advanced programming in the Unix environment, W.R.Stevens, Pearson education.

## References

1. Operating Systems – Internals and Design Principles Stallings, Fifth Edition–2005, Pearson Education/PHI
2. Operating System A Design Approach-Crowley,TMH.
3. Modern Operating Systems, Andrew S Tanenbaum 2nd edition, Pearson/PHI
4. Unix programming environment, Kernighan and Pike, PHI. / Pearson Education
5. Unix Internals The New Frontiers, U.Vahalia, Pearson Education

## 2036772-DATABASE MANAGEMENT SYSTEMS LAB

**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**
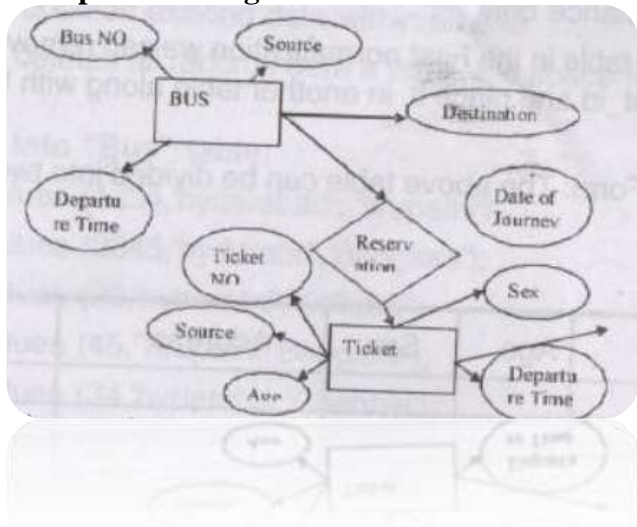


Note: The students is required to submit a document by drawing the E-R Diagram to the lab teacher.

**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 |
|------|-----|-----|---------|-----------|-------------|
|      |     |     |         |           |             |
|      |     |     |         |           |             |

| 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.

# 2036773-PYTHON PROGRAMMING LAB

**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
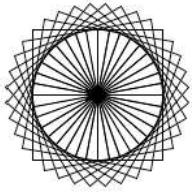
**TEXT BOOKS**:

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.

**2030025: Gender Sensitization & Community Participation**
**B.Tech. II Year I -Semester**                                                     **L T  P C**
                                                                                     **2  0 0 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 andConditions
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- FurtherReading:
‒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: NewForums 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, DuggiralaVasanta, Rama Melkote, Vasudha Nagaraj, Asma Rasheed, GoguShyamala, Deepa Sreenivas and Susie Tharuand 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

## 2046705-DISCRETE MATHAMATICS

**B.Tech. II Year II Sem.**

<div align="right">

**L  T  P  C**
**3  0  0  3**

</div>

### 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 calcul**us.**

### 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 -Semester**                                     **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.

## 2046706-JAVA PROGRAMMING

**B.Tech. II Year II Sem.**
**C**

**L  T  P**

**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 skleton, 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.

# 2046707-INTRODUCTION TO DATA SCIENCE

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

**Prerequisites**

        Basics concepts of Python and Database management systems.

**Course Objective:**

1. To get knowledge on data science process.
2. To get basic understanding of how to apply python to data science.
3. To understand the concepts of Statistical Natural Language Processing for Sentiment Analysis

**Course Outcome:**

**The students should be able to**

1. Understand the process of data science.
2. Use Python libraries for data preparation and processing.
3. Understand the Statistical inference.
4. Describe network analysis and Recommender systems.

## UNIT-I

Data science in a big data world: Benefits and uses of data science and big data, Facets of Data, The data science process, Big data eco system and data science.
Data science process: Defining research goals and creating a project charter, Retrieving data, Cleansing, integrating, and transforming data, exploratory data analysis, Build the models, presenting findings and building applications on top of them.

## UNIT-II

Toolboxes for Data Scientists: Python Libraries for Data Scientists , Get Started with Python for Data Scientists.
Descriptive Statistics: Data Preparation. Exploratory Data Analysis, Estimation

## UNIT-III

Statistical Inference: The Frequentist Approach, Measuring the Variability in Estimates., Hypothesis Testing Supervised Learning: Learning Curves , Training, Validation and Test, Regression Analysis: Linear Regression, Multiple Linear Regression, Logistic Regression

## UNIT-IV

Network Analysis : Basic Definitions in Graphs, Social Network Analysis, Centrality, Ego-Networks, Community Detection
Recommender Systems: How Do Recommender Systems Work? Modeling User Preferences, Evaluating Recommenders, **Practical case**: MovieLens Dataset

## UNIT-V

Statistical Natural Language Processing for Sentiment Analysis:   Data Cleaning, Text Representation, Practical Cases, Parallel Computing.: Architecture, Multicore Programming, Distributed Computing.
A Real time Application: New York Taxi Trips

**TEXT BOOK:**

i)      Introduction to data science    by  davy cielen, arno d. B. Meysman, mohamed ali manning shelter island

ii)     "Introduction to Data Science" A Python Approach to Concepts, Techniques and Applications By  Laura Igual · Santi Seguí    Spinger

**REFERENCE BOOKS**

1. Michael Berthold, David J. Hand, "Intelligent Data Analysis", Springer, 2007.

2. Anand Rajaraman and Jeffrey David Ullman, "Mining of Massive Datasets", Cambridge University Press, 2012.

3. Bill Franks, "Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics", John Wiley & sons, 2012.

4. Jiawei Han, Micheline Kamber "Data Mining Concepts and Techniques", Second Edition, Elsevier, Reprinted 2008.

5. Rachel Schutt, Cathy O'Neil, "Doing Data Science", O'Reilly Publishers, 2013

## 2046708-DATA WAREHOUSING AND DATA MINING

**B.TECH II Year II Sem.**                                         **L T P C**
                                                                         **2 0 0 2**

**Course Objective**:
1. Be familiar with mathematical foundations of data mining tools.
2. Understand and implement classical models and algorithms in data warehouses and data mining.
3. Characterize the kinds of patterns that can be discovered by association rule mining, classification and clustering.
4. Master data mining techniques in various applications like social, scientific and environmental context.
5. Develop skill in selecting the appropriate data mining algorithm for solving practical problems.

**Course Outcome:**
**The students should be able to**

1. Understand the functionality of the various data mining and data warehousing component.
2. Appreciate the strengths and limitations of various data mining and data warehousing models.
3. Explain the analyzing techniques of various data.
4. Describe different methodologies used in data mining and data ware housing.
5. Compare different approaches of data ware housing and data mining with various technologies. Evaluating

## UNIT – I
**Data Warehouse basic concepts:** what is a Data warehouse, Differences between operational data base systems and data Warehouse, Architecture, Models, Extraction-Transformation-Loading, Metadata repository.
**Data Warehousing Modeling**, Data Cube, star and snow-Flake Schema, Fact Constellation, Dimensions, Measures, OLAP Operations.

## UNIT -II
**Introduction to Data Mining:** Introduction, What is Data Mining, KDD
**Data Preprocessing-** Data processing overview, Data Cleaning, Data integration, Data Reduction, Data Transformation and Data Discretization.

## UNIT -III
**Classification:** Preliminaries**,** General Approaches to solving a classification problem, Decision trees-Induction, Decision Tree Construction, nearest neighbor classifier, Bayesian classifier, ANN, SVM.

## UNIT – IV

**Association Rules:** Problem Definition, Frequent Item Set Generation, The APRIORI Principle, Rule Generation- Rule Generation in APRIORI Algorithm, Compact Representation of Frequent Item Set-Maximal Frequent Item Set, Closed Frequent Item Set, FP-Growth Algorithms.

**UNIT – V**

**Cluster Analysis:** Clustering overview, different types of clustering, K-Means Algorithm-K-Means Additional Issues, PAM Algorithm, Agglomerative Hierarchical Clustering Algorithm, Specific techniques, Key Issues in Hierarchical Clustering, Strengths and weakness, DBSCAN, Cluster evolution.

**TEXT BOOK:**
1. Data Mining-Concepts and Techniques- Jiawei Han, Micheline Kamber, Morgan Kaufmann Publishers, Elsevier, 2 Edition, 2006. (UNIT 1 and 2)
2. Introduction to Data Mining, Pang-Ning Tan, Vipin Kumar, Michael Steinbanch, Pearson Education. (. (UNIT 3,4 and 5)

**REFERENCES:**
1. Data Mining Techniques, Arun K Pujari, 3rd Edition, Universities Press.
2. Data Ware Housing Fundamentals, PualrajPonnaiah, Wiley Student Edition.
3. The Data Ware House Life Cycle Toolkit- Ralph Kimball, Wiley Student Edition.
4. Data Mining, VikaramPudi, P Radha Krishna, Oxford University.

# 2040271: BASIC ELECTRICAL ENGINEERING LAB

**B.Tech. II Year II Semester.**                                                **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 electricalparameters
- 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 RLC circuit.
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 aSingle PhaseTransformer.
8. Load Test on Single Phase Transformer (Calculate Efficiency and Regulation)
9. Three Phase Transformer: Verification of Relationship between Voltages and 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 InductionMotor.
14. Torque-Speed Characteristics of a Three-phase InductionMotor.
15. No-Load Characteristics of a Three-phaseAlternator.

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

**0 0 2 1**

**Course Outcomes:**
The students should be able to
1. Get the acquaintance to WEKA tool
2. Competent to preprocess the data for mining
3. Proficient in generating association rules
4. Build various classification models
5. Realise clusters from the available data

**LIST OF EXPERIMENTS:**

1. Basics of WEKA tool
    a. Investigate the Application interfaces.
    b. Explore the default datasets.

2. Pre-process a given dataset based on the following:
    a. Attribute Selection
    b. Handling Missing Values
    c. Discretization
    d. Eliminating Outliers

3. Pre-process a given dataset based on the following:
    a. Discretization
    b. Eliminating Outliers

4. Create a dataset in ARFF (Attribute-Relation File Format) for any given dataset and perform Market-Basket Analysis.

5. Generate Association Rules using the Apriori algorithm.

6. Generate Association Rules using the FP-Growth algorithm.

7. Build a Decision Tree by using ID3 algorithm.

8. Build various Regression models.

9. Cluster the given dataset by using the k-Means Clustering algorithm and visualize the cluster mean values and standard  deviation of dataset attributes.

10. Cluster the given dataset by using the DBSCAN Clustering  algorithm.

11. Cluster the given dataset by using the Expectation Maximization clustering algorithm.

**TEXT BOOKS:**

1. JlaweiHan and Kamber,"Data Mining Concepts and  Techniques", Third Edition, Elsevier, 2011.

**REFERENCE:**
1. Ian H.Witten,EibeFank, Mark A Hall, "Data Mining Practical Machine Learning Tools and Techniques", Third edition,Morgan Kaufmann Publishers, 2011.

**WEB REFERENCE:**
www.cs.waikato.ac.nz/ml/weka/downloading.html

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

**Prerequisites:**

Programming for problem solving lab

**Course Objectives:**
1. To write programs using abstract classes.
2. To write programs for solving real world problems using java collection frame work.
3. To write multithreaded programs.
4. To write GUI programs using swing controls in Java.

**Course Outcomes:**
.   **The students should be able to**
1. Write programs for solving real world problems using java collection frame work.
2. Write programs using abstract classes.
3. Write multithreaded programs.
4. Write GUI programs using swing controls in Java.

## List of Experiments

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.

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 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.

4. a) 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 Num 2 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.
   b).Write a java program to create user defined exception class and test this class.

5. a) Write a Java program to list all the files in a directory including the files present in all its subdirectories.

b) Write a java program that displays the number of characters, lines and words in a text file.

6.  a) 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.
    b) Write a Java program that correctly implements the producer – consumer problem using the concept of inter thread communication.

7.  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.

8.  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).

9.  a) 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).
    b) Write a java program to demonstrate the key event handlers.

10. a) Develop an applet in Java that displays a simple message.
    b) Develop an applet in Java that receives an integer in one text field, and computes its factorial
    Value and returns it in another text field, when the button named "Compute" is clicked.

11. 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.

12. 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.

13. Develop Swing application which uses JList, JTree, JTable, JTabbedPane and JScrollPane.

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

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

**REFERENCE BOOKS**:
1. Java for Programmers, P. J. Deitel and H. M. Deitel, 10th Edition Pearson education.
2. Thinking in Java, Bruce Eckel, Pearson Education.
3. Java Programming, D. S. Malik and P. S. Nair, Cengage Learning.
4. Core Java, Volume 1, 9th edition, Cay S. Horstmann and G Cornell,

**2040023: CONSTITUTION OF INDIA**

**B.Tech. II Year II Semester.**                                   **L T P C**
                                                                   **2 0 0 0**

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 political ideology and result of historical struggles against arbitrary use of sovereign power by state. The historic revolutions in France, England, America and particularly European Renaissance and Reformation 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. The seamendments 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 ofpeople. It has been protecting not only basic ideals of the Constitution but also strengthened the samethrough progressive interpretations of the text ofthe Constitution. The judicial activism of the SupremeCourt of India and its historic contributions has been recognized throughout the world and it graduallymade 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