

COURSE STRUCTURE (R20)**O- SEMESTER**

Universal Human Values 1: Student Induction Program

3 Weeks

I – SEMESTER

| S. No. | Course Code | Course Title | Contact hours/week | | | Credits | Scheme of Valuation | | |
|---------------------------|-------------|----------------------------------------------|--------------------|----------|-----------|-------------|---------------------|------------|------------|
| | | | L | T | P | | CIE | SEE | Total |
| Theory Courses | | | | | | | | | |
| 1 | 20FE01 | Professional Communication-I | 2 | 0 | 0 | 2 | 30 | 70 | 100 |
| 2 | 20FE03 | Differential Equations | 2 | 1 | 0 | 3 | 30 | 70 | 100 |
| 3 | 20FE06 | Engineering Chemistry | 3 | 0 | 0 | 3 | 30 | 70 | 100 |
| 4 | 20CS01 | Programming for Problem Solving using C | 3 | 0 | 0 | 3 | 30 | 70 | 100 |
| 5 | 20EE02 | Basic Electrical and Electronics Engineering | 3 | 0 | 0 | 3 | 30 | 70 | 100 |
| Laboratory Courses | | | | | | | | | |
| 6 | 20FE51 | Professional Communication Skills Lab | 0 | 0 | 2 | 1 | 15 | 35 | 50 |
| 7 | 20FE53 | Engineering Chemistry Lab | 0 | 0 | 3 | 1.5 | 15 | 35 | 50 |
| 8 | 20CS51 | Programming for Problem Solving using C Lab | 0 | 0 | 3 | 1.5 | 15 | 35 | 50 |
| 9 | 20ME53 | Computer Aided Engineering Drawing | 0 | 0 | 3 | 1.5 | 15 | 35 | 50 |
| Total | | | 13 | 1 | 11 | 19.5 | 210 | 490 | 700 |

II - SEMESTER

| S. No. | Course Code | Course Title | Contact hours/week | | | Credits | Scheme of Valuation | | |
|---------------------------|-------------|----------------------------------------------|--------------------|----------|-----------|-------------|---------------------|------------|------------|
| | | | L | T | P | | CIE | SEE | Total |
| Theory Courses | | | | | | | | | |
| 1 | 20FE02 | Professional Communication-II | 2 | 0 | 0 | 2 | 30 | 70 | 100 |
| 2 | 20FE04 | Linear Algebra and Transformation Techniques | 2 | 1 | 0 | 3 | 30 | 70 | 100 |
| 3 | 20FE07 | Applied Physics | 2 | 1 | 0 | 3 | 30 | 70 | 100 |
| 4 | 20CS05 | Python Programming | 3 | 0 | 0 | 3 | 30 | 70 | 100 |
| 5 | 20CS03 | Data Structures | 3 | 0 | 0 | 3 | 30 | 70 | 100 |
| 6 | 20MC01 | Constitution of India | 2 | 0 | 0 | 0 | 30 | 70 | 100 |
| Laboratory Courses | | | | | | | | | |
| 7 | 20FE54 | Applied Physics Lab | 0 | 0 | 3 | 1.5 | 15 | 35 | 50 |
| 8 | 20CS54 | Python Programming Lab | 0 | 0 | 3 | 1.5 | 15 | 35 | 50 |
| 9 | 20CS53 | Data Structures Lab | 0 | 0 | 3 | 1.5 | 15 | 35 | 50 |
| 10 | 20IT52 | Mathematical Applications Lab | 0 | 0 | 2 | 1 | 15 | 35 | 50 |
| Total | | | 14 | 2 | 11 | 19.5 | 240 | 560 | 800 |

III SEMESTER

| S. No. | Course code | Course Title | Contact hours/week | | | Credits | Scheme of Valuation | | |
|---------------------------|-------------|--------------------------------------------------------------------------------|--------------------|----------|-----------|-------------|---------------------|------------|------------|
| | | | L | T | P | | CIE | SEE | Total |
| Theory Courses | | | | | | | | | |
| 1 | 20FE09 | Probability & Statistics | 3 | 0 | 0 | 3 | 30 | 70 | 100 |
| 2 | 20CS04 | Discrete Mathematical Structures | 3 | 0 | 0 | 3 | 30 | 70 | 100 |
| 3 | 20CS07 | Database Management Systems | 3 | 0 | 0 | 3 | 30 | 70 | 100 |
| 4 | 20CS08 | Computer Organization | 3 | 0 | 0 | 3 | 30 | 70 | 100 |
| 5 | 20CS09 | Object Oriented Programming | 3 | 0 | 0 | 3 | 30 | 70 | 100 |
| Laboratory Courses | | | | | | | | | |
| 6 | 20CS56 | Database Management Systems Lab | 0 | 0 | 3 | 1.5 | 15 | 35 | 50 |
| 7 | 20CS57 | Object Oriented Programming Lab | 0 | 0 | 3 | 1.5 | 15 | 35 | 50 |
| 8 | 20IT53 | R Programming Lab | 0 | 0 | 3 | 1.5 | 15 | 35 | 50 |
| 9 | 20CSS1 | Web Application Development using Full Stack - Module-I (Frontend Development) | 1 | 0 | 2 | 2 | - | 50 | 50 |
| Total | | | 16 | 0 | 11 | 21.5 | 195 | 505 | 700 |

IV SEMESTER

| S. No. | Course code | Course Title | Contact hours/week | | | Credits | Scheme of Valuation | | |
|---------------------------|-------------|-------------------------------------------------------------------------------|--------------------|----------|-----------|-------------|---------------------|------------|------------|
| | | | L | T | P | | CIE | SEE | Total |
| Theory Courses | | | | | | | | | |
| 1 | 20CS06 | Design & Analysis of Algorithms | 3 | 0 | 0 | 3 | 30 | 70 | 100 |
| 2 | 20CS10 | Data Warehousing & Data Mining | 3 | 0 | 0 | 3 | 30 | 70 | 100 |
| 3 | 20CS11 | Operating Systems | 3 | 0 | 0 | 3 | 30 | 70 | 100 |
| 4 | 20IT01 | Software Engineering | 3 | 0 | 0 | 3 | 30 | 70 | 100 |
| 5 | 20HS01 | Universal Human Values 2: Understanding Harmony | 3 | 0 | 0 | 3 | 30 | 70 | 100 |
| 6 | 20MC02 | Environmental Science | 2 | 0 | 0 | 0 | 30 | 70 | 100 |
| Laboratory Courses | | | | | | | | | |
| 7 | 20CS58 | Data Mining using Python Lab | 0 | 0 | 3 | 1.5 | 15 | 35 | 50 |
| 8 | 20IT54 | Operating Systems and Linux Internals Lab | 0 | 0 | 3 | 1.5 | 15 | 35 | 50 |
| 9 | 20IT55 | Software Engineering Lab | 0 | 0 | 3 | 1.5 | 15 | 35 | 50 |
| 10 | 20CSS2 | Web Application Development using Full Stack- Module-II (Backend Development) | 1 | 0 | 2 | 2 | - | 50 | 50 |
| Total | | | 18 | 0 | 11 | 21.5 | 225 | 575 | 800 |
| Honors/Minor Courses | | | 3 | 1 | 0 | 4 | 30 | 70 | 100 |

B.Tech. (I Sem.) 20FE01 - PROFESSIONAL COMMUNICATION - I

| L | T | P | Cr. |
|---|---|---|-----|
| 2 | 0 | 0 | 2 |

Pre-requisites: Nil

Course Educational Objectives: To improve English language proficiency of the students on various aspects like vocabulary, grammar, communication skills, listening skills, Reading & Writing skills.

Course Outcomes: At the end of the course, the student will be able to

- CO1** : Write sentences and paragraphs using proper grammatical structures and word forms(**Remember – L1**).
- CO2** : Comprehend the given text by employing suitable strategies for skimming and scanning and draw inferences(**Understand –L2**).
- CO3** : Write summaries of reading texts using correct tense forms & appropriate structures(**Remember – L1**)
- CO4** : Write Formal Letters; Memos & E-Mails (**Apply – L3**).
- CO5** : Edit the sentences/short texts by identifying basic errors of grammar/vocabulary/syntax(**Understand – L2**).

Unit - I

Exploration - ‘A Proposal to Girdle the Earth – Nellie Bly’; Reading: Skimming for main idea; Scanning for specific information; Grammar & Vocabulary: Content Words; Function Words; Word Forms: verbs, nouns, adjectives and adverbs; Nouns: Countable and Uncountable, Singular and Plural forms; Wh - Questions; Word Order in Sentences; Writing: Paragraph Analysis; Paragraph Writing; Punctuation and Capital Letters

Unit – II

On Campus- ‘The District School as it Was by One Who Went to it – Warren Burton’; Reading: Identifying Sequence of Ideas; Grammar & Vocabulary: Cohesive Devices: Linkers/signposts/Transition signals, Synonyms, Meanings of Words/Phrases in the context; Writing: Memo Drafting.

Unit – III

Working Together-‘The Future of Work’

Reading: Making basic inferences; Strategies to use text clues for comprehension; Summarizing; Grammar & Vocabulary: Verbs: Tenses; Reporting Verbs for Academic Purpose; Writing: Rephrasing what is read; Avoiding redundancies and repetitions Abstract Writing/ Summarizing.

Unit – IV

‘**A.P.J.AbdulKalam**’; Grammar & Vocabulary: Direct & Indirect Speech; articles and their Omission; Writing: E-Mail Drafting.

Unit – V

‘**C.V.Raman**’; Grammar & Vocabulary: Subject-Verb Agreement; Prepositions; Writing: Formal Letter Writing.

TEXTBOOKS:

1. Prabhavati. Y & et al, “English All Round – Communication Skills for Undergraduate Learners”, Orient Black Swan, Hyderabad, 2019.
2. “Panorama – A Course on Reading”, A collection of prose selections, Oxford University Press, New Delhi, 2016.

Reference Books:

1. Swan,M., “Practical English Usage”, Oxford University Press, 2016.
2. Kumar,S and Latha, P, “Communication Skills”, Oxford University Press, 2018.
3. Rizvi Ashraf M., “Effective Technical Communication”, Tata Mc Graw Hill, New Delhi, 2008.
4. BaradwajKumkum, “Professional Communication”, I.K.International Publishing House Pvt.Lt., New Delhi, 2008.
5. Wood,F.T., “Remedial English Grammar”, Macmillan, 2007.

B.Tech. (I Sem.)

20FE03 - DIFFERENTIAL EQUATIONS

| L | T | P | Cr. |
|---|---|---|-----|
| 2 | 1 | 0 | 3 |

Pre-requisites: Nil

Course Educational Objective: The objective of this course is to introduce the first order and higher order differential equations, functions of several variables. The students also learn solving of first order partial differential equations.

Course Outcomes: At the end of this course, the student will be able to

- CO1:** Apply first order and first-degree differential equations to find orthogonal trajectories (**Apply – L3**).
- CO2:** Distinguish between the structure and methodology of solving higher order differential equations with constant coefficients (**Understand – L2**).
- CO3:** Apply various Numerical methods to solve initial value problem (**Apply – L3**).
- CO4:** Generate the infinite series for continuous functions and investigate the functional dependence (**Understand – L2**).
- CO5:** Solve partial differential equations using Lagrange's method (**Apply – L3**).

UNIT –I**Differential Equations of First Order and First Degree**

Differential equations of first order and first degree –Exact and Non Exact differential Equations, Applications of differential equations – Orthogonal Trajectories.

UNIT –II**Linear Differential Equations of Higher Order**

Homogeneous and Non Homogeneous Linear differential equations of second and higher order with constant coefficients with R.H.S. functions e^{ax} , $\sin(ax+b)$, $\cos(ax+b)$, x^m , $e^{ax}V(x)$, $xV(x)$, Method of variation of parameters.

UNIT – III**Numerical solution of Ordinary Differential Equations**

Numerical solution of Ordinary Differential equations, Solution by Taylor's series - Picard's Method of successive approximations.

Euler's Method - Runge- Kutta Methods.

UNIT –IV**Functions of several variables**

Generalized Mean Value Theorem (without proof), Maclaurin's series, Functions of several variables, Jacobians (Cartesian and polar coordinates), Functional dependence. Maxima and Minima of function with two variables.

UNIT – V**Partial Differential Equations**

Formation of Partial Differential Equations by elimination of arbitrary constants and arbitrary functions. Solution of first order and first degree linear partial differential equation – Lagrange’s method.

TEXTBOOKS:

1. B.S. Grewal, “*Higher Engineering Mathematics*”, 42nd Edition, Khanna Publishers, New Delhi, 2012.
2. B. V. Ramana, “*Higher Engineering Mathematics*”, 1st Edition, TMH Publications, New Delhi, 2010

Reference Books:

1. M. D. Greenberg, “*Advanced Engineering Mathematics*”, 2nd Edition, TMH Publications, New Delhi, 2011.
2. Erwin Kreyszig, “*Advanced Engineering Mathematics*”, 8th Edition, John Wiley & sons, New Delhi, 2011.
3. W.E. Boyce and R. C. Diprima, “*Elementary Differential Equations*”, 7th Edition, John Wiley & sons, New Delhi, 2011.
4. S. S. Sastry, “*Introductory Methods of Numerical Analysis*”, 5th Edition, PHI Learning Private Limited, New Delhi, 2012.

| L | T | P | Cr. |
|---|---|---|-----|
| 3 | 0 | 0 | 3 |

B.Tech (I Sem)

20FE06 - ENGINEERING CHEMISTRY**Pre-requisites:** Nil

Course Educational Objectives: It enables the students to understand the fundamental concepts of chemistry and to provide them with the knowledge of industrial problems and finding the solutions. It helps to strengthen the basic concepts of electrochemistry, corrosion, nanotechnology, polymers, liquid crystals and analytical techniques.

Course Outcomes: At the end of the course, students will be able to,

- CO1:** Apply Nernst Equation for calculating electrode cell potentials and compare batteries in different applications (**Apply – L3**).
- CO2:** Apply principles of corrosion for design and effective maintenance of various equipment (**Apply – L3**).
- CO3:** Analyse the suitability of advanced materials like nano materials in electronics and medicine (**Understand – L2**).
- CO4:** Identify the importance of liquid crystals, polymers in advanced technologies (**Understand – L2**).
- CO5:** Apply the principles of analytical techniques in chemical analysis (**Apply – L3**).

UNIT – I**Electro Chemistry & Batteries**

Types of Electrodes - Calomel Electrode, Glass Electrode, Calculation of EMF of Cell, Applications of Nernst Equation & Electro chemical Series, Batteries -Lead-acid Battery, Lithium ion Battery, H₂ – O₂ Fuel Cell, Mg - Cu reserve battery.

UNIT – II**Science of Corrosion**

Dry Corrosion (Direct Chemical corrosion) - Types of dry corrosion-oxidative corrosion, Pilling Bed worth rule, corrosion by other gases and liquid metal corrosion; Wet Corrosion (Electro Chemical corrosion) - Mechanism- oxygen absorption, hydrogen evolution, types of wet corrosion, Galvanic Corrosion, Concentration Cell Corrosion, passivity and Galvanic series; Factors Influencing Corrosion - Nature of metal (Purity, position in galvanic series, relative area of cathode & anode, nature of surface film) and nature of environment (temperature, humidity, atmospheric pollution and nature of ions in the medium); Control of Corrosion: Cathodic Protection - Sacrificial anode and impressed current methods, electro plating and metal cladding.

UNIT – III**Chemistry of Engineering Materials**

Nano Materials - Extraordinary changes observed at nano size of materials and reasons, types of nano-materials, Gas-Phase Synthesis of nanomaterials, Applications; Materials in Electronic devices: Very brief note on raw materials that make IC units of CPU, GPU, RAM, PCBs, hard disks and other electronic devices with special reference to polymers;

Molecular Switches - Characteristics of Molecular motors and machines, Rotaxanes and Catenanes as artificial molecular machines, prototypes – linear motions in rotaxanes, an acid-base controlled molecular shuttle, molecular elevator, automated light-powered molecular motor.

UNIT – IV

Liquid Crystals & Polymers

Liquid crystals - Identification and structural aspects of molecules to form liquid crystals; Classification of liquid crystals - Thermo tropic liquid crystals and types, lyotropic liquid crystals. Mechanism of working of liquid crystals and applications; Polymers - Differences between thermoplasts and thermosets, Types of polymerization with examples; Plastics - Preparation properties and engineering applications of P.M.M.A, Teflon, Polycarbonate; Rubbers - Structure of raw rubber and vulcanized rubber, Preparation properties and engineering applications of Polyurethane, Buna-S, conducting polymers; Bio-degradable polymers - PLA & PGA (Polylactic Acid and Polyglycolic Acid).

UNIT – V

Analytical Techniques

Types of analysis; Physical analysis: Analysis of physical characteristics; Chemical analysis: Gravimetric and volumetric analysis (basic concept only); Instrumental analysis: Electro analytical techniques – Introduction; Conductometric techniques: strong acid-strong base and strong acid-weak base, weak acid -strong base and weak acid -weak base & advantages; Potentiometric techniques: Acid-base and oxidation-reduction titrations-advantages; Colorimetric techniques: Principle and determination of iron by using thiocyanate as a reagent.

TEXTBOOKS

1. Shikha Agarwal, “A Text book of Engineering Chemistry”, Cambridge University Press, New Delhi, 1st Edition, 2015.
2. Jain, Jain, “A textbook of Engineering Chemistry”, Dhanpat Rai Publishing Company, New Delhi, 16th Edition, 2015.

REFERENCE BOOKS

1. Shashi Chawla, “A Text book of Engineering Chemistry”, Dhanpat Rai Publishing Company, New Delhi, 3rd Edition, 2003.
2. S.S. Dara, S.S. Umare, “A Text book of Engineering Chemistry”, S. Chand Publications, New Delhi, 12th Edition, 2010.
3. Prasanta Rath, B. Rama Devi, Ch. Venkata Ramana Reddy, Subhendu Chakroborty, “Engineering Chemistry”, Cengage Learning India, 1st Edition, 2019.

B.Tech. (I Sem.)

20CS01 - PROGRAMMING FOR PROBLEM SOLVING USING C

| L | T | P | Cr. |
|---|---|---|-----|
| 3 | 0 | 0 | 3 |

Pre-requisite : Nil

Course Educational Objective: The Objective of the course is to make learn the basic elements of C programming, control structures, derived data types, Modular programming, user defined structures, basics of files and its I/O operations.

Course Outcomes: At the end of this course, the student will be able to

- CO1:** Familiar with syntax and semantics of the basic programming language constructs. (**Understand - L2**)
- CO2:** Construct derived data types like arrays in solving problem. (**Apply - L3**)
- CO3:** Decompose a problem into modules and reconstruct it using various ways of user-defined functions. (**Apply - L3**)
- CO4:** Use user-defined data types like structures and unions and its applications to solve problems. (**Apply - L3**)
- CO5:** Discuss various file I/O operations and its application. (**Understand - L2**)

UNIT – I

Introduction to Problem solving through C-Programming: Problem Specification, Algorithm / pseudo code, flowchart, examples.

C-Programming: Structure of C program, identifiers, basic data types and sizes, Constants, variables, Input-output statements, A sample c program, operators, expressions, type conversions, conditional expressions, precedence of operators and order of evaluation.

Control statements: if, if else, else if ladder and switch statements, while, do-while and for statements, break, continue, go to and labels.

UNIT – II

Arrays- concept, declaration, definition, accessing elements, storing elements, two dimensional and multi-dimensional arrays.

Character Arrays: declaration, initialization, reading, writing strings, string handling functions, pre-processor Directives, and macros.

Applications of Arrays: Linear search, Binary search, Bubble Sort.

UNIT – III

Pointers- concepts, declaring & initialization of pointer variables, pointer expressions, pointer arithmetic, pointers and arrays, pointers and character arrays, pointers to pointers.

Functions: basics, category of functions, parameter passing techniques, recursive functions-comparison with Iteration, Functions with arrays, Standard library functions, dynamic memory management functions, command line arguments.

Storage classes - auto, register, static and extern.

UNIT – IV

Derived types- structures- declaration, definition, and initialization of structures, accessing structures, nested structures, arrays of structures, structures and functions, pointers to structures, self-referential structures, unions, typedef.

UNIT – V

Files – concept of a file, text files and binary files, streams, standard I/O, Formatted I/O, file I/O operations, error handling.

Textbook:

1. ReemaThareja, Programming in C, Oxford University Press, 2nd Edition, 2015

Reference books:

1. Jeri R.Hanly, Elliot B.Koffman, Problem Solving and Program Design in C, Pearson Publishers, 7th Edition, 2013
2. E Balagurusamy, Computer Programming, McGraw Hill Education, 8th Edition.
3. C: The Complete Reference, McGraw Hall Education,4th Edition.
4. PradeepDey, Manas Ghosh, Programming in C, Oxford University Press, 2nd Edition, 2011.
5. Stephen G.Kochan, Programming in C, Pearson Education, 3rd Edition, 2005.

**20EE02 - BASIC ELECTRICAL AND ELECTRONICS
ENGINEERING**

B.Tech. (ISem.)

| L | T | P | Cr. |
|----------|----------|----------|------------|
| 3 | 0 | 0 | 3 |

Prerequisite: Physics

Course Educational Objective: This course enables student to illustrate the basics of applied electricity and electronics.

Course Outcomes: At the end of the course, the student will be able to:

- CO1:** Apply network reduction techniques to simplify electrical circuits (**Apply – L3**).
- CO2:** Illustrate the working principle of DC machines and transformers (**Understand – L2**).
- CO3:** Understand V-I characteristics of semiconductor devices (**Understand – L2**).
- CO4:** Illustrate the configuration of Transistors and their applications (**Understand – L2**).

UNIT – I: Electrical Circuit Fundamentals

Basic definitions, Types of elements-active and passive, Ohm’s Law, Kirchhoff’s Laws-Network reduction techniques- series, parallel, star to delta, delta to star transformations, source transformation (for resistive networks), mesh analysis, nodal analysis (Basic problems).

UNIT – II: DC Network Theorems and AC Fundamentals

Theorems-Superposition, Thevenin’s, Norton’s and Maximum Power Transfer (Basic problems in DC excitation only)

Peak, R.M.S, average, instantaneous values, form factor and peak factor– periodic waveforms – Phase and Phase difference –concepts of reactance, impedance, susceptance and admittance, real, reactive and apparent powers, Power Factor- resonance-bandwidth-quality factor.

UNIT – III: DC Machine Fundamentals and Single-Phase Transformers

DC generator principle, constructional details, emf equation, types of generators (Theory only).

DC motor principle, Back emf, types of motor (Theory only).

Construction and Principle of operation of single-phase transformers-Emf equation

UNIT-IV: P-N Junction Diode and Zener Diode

P-N Junction Diode: Operation and V-I characteristics of PN junction diode, Rectifiers-Half Wave Rectifier, Full Wave Rectifier-Bridge type, Zener Diode-Voltage Regulator.

UNIT – V: Transistors

Construction, Principle of Operation, Symbol, CB, CE configurations, JFET, MOSFET and application of transistor as an amplifier (Theory only).

TEXTBOOKS:

1. A.Sudhakar and Shyammohan S Palli, “Electrical Circuits” Tata McGraw-Hill, 3rd Edition.2017
2. M.S.Sukhija, T.K.Nagsarkar, “Basic Electrical and Electronics Engineering”, Oxford University Press, 2016 Edition.

REFERENCE BOOK(S):

1. Kothari and Nagarath, “Basic Electrical Engineering”, TMH Publications, 3rd Edition.2013
2. G.S.N.Raju, “Electronic Devices and Circuits”, I.K.International.2006

B.Tech. (I Sem.)

**20FE51 - PROFESSIONAL COMMUNICATION
SKILLS LAB**

| L | T | P | Cr. |
|---|---|---|-----|
| 0 | 0 | 2 | 1 |

Pre-requisites : Nil

Course Educational Objective: To improve the proficiency of students in English with an emphasis on better communication in formal and informal situations; Develop speaking skills required for expressing their knowledge and abilities and to face interviews with confidence.

Course Outcomes: At the end of this course, the student will be able to

- CO1:** Introduce oneself and others using appropriate language and details (**Understand– L2**).
- CO2:** Comprehend short talks and speak clearly on a specific topic using error free English (**Understand– L2**).
- CO3:** Report effectively after participating in informal discussions ethically (**Remember– L1**).
- CO4:** Interpret data aptly, ethically & make oral presentations (**Apply –L3**).

Syllabus: Professional Communication Lab (PCS) shall have two parts:

- **Computer Assisted Language Learning (CALL) Lab** for 60 students with 60 systems, LAN facility and English language software for self- study by learners.
- **Interactive Communication Skills (ICS) Lab.** with movable chairs and audio-visual aids with a P.A System, a T. V., a digital stereo – audio & video system and camcorder etc.

Exercise – I

CALL Lab: Understand - Sentence structure.

ICS Lab: Practice - Listening: Identifying the topic, the context and specific information
Speaking: Introducing oneself and others.

Exercise – II

CALL Lab: Understand - Framing questions.

ICS Lab: Practice - Listening: Answering a series of questions about main idea and supporting ideas after listening to audio text.
Speaking: Discussing in pairs/small groups on specific topics; Delivering short structured talks using suitable cohesive devices (JAM)

Exercise – III

CALL Lab: Understand - Comprehension practice – Strategies for Effective Communication

ICS Lab: Practice - Listening: Listening for global comprehension and summarizing
Speaking: Discussing specific topics in pairs/small groups, reporting what is discussed

Exercise – IV

CALL Lab: Understand- Features of Good Conversation – Strategies for Effective Communication.

ICS Lab: Practice -Listening: making predictions while listening to conversations/transactional dialogues with/without video Speaking: Role – plays – formal & informal – asking for and giving information/directions/instructions/suggestions

Exercise – V

CALL Lab: Understand - Features of Good Presentation, Methodology of Group Discussion

ICS Lab:Practice - Introduction to Group Discussions.

Listening: Answering questions identifying key terms and understanding concepts.

Speaking: Formal Oral & Poster presentations on topics from academic contexts without the use of PPT.

Lab Manual:

1. Prabhavati. Y & et al, “English All Round – Communication Skills for Undergraduate Learners”, Orient Black Swan, Hyderabad, 2019.

Suggested Software:

1. Digital Mentor: Globarena, Hyderabad, 2005
2. Sky Pronunciation Suite: Young India Films, Chennai, 2009
3. Mastering English in Vocabulary, Grammar, Spelling, Composition, Dorling Kindersley,USA,2001
4. Dorling Kindersley Series of Grammar, Punctuation, Composition, USA, 2001
5. Oxford Talking Dictionary, The Learning Company, USA, 2002
6. Learning to Speak English - 4 CDs. The Learning Company,USA,2002
7. Cambridge Advanced Learners English Dictionary (CD). Cambridge University Press, New Delhi,2008.

B.Tech. (I Sem.)

20FE53 - ENGINEERING CHEMISTRY LAB

| L | T | P | Cr. |
|---|---|---|-----|
| 0 | 0 | 3 | 1.5 |

Pre-requisites: Nil

Course Educational Objectives: This course enables the students to analyze water samples and perform different types of volumetric titrations. It provides them with an overview of preparation of polymers and analytical techniques.

Course Outcomes: At the end of the course, the student shall be able to

- CO 1: Assess alkalinity of water based on the procedure given (**Understand – L2**).
- CO 2: Distinguish different types of titrations in volumetric analysis after performing the experiments listed in the syllabus (**Understand – L2**)
- CO 3: Acquire practical knowledge related to preparation of polymers(**Understand – L2**).
- CO 4: Exhibit skills in performing experiments based on theoretical fundamental(**Understand – L2**).

List of Experiments

(Any of the 10 experiments are required to be conducted)

Model Experiment

1. Determination of HCl using standard Na_2CO_3 solution.

Water Analysis

2. Determination of alkalinity of water sample.

Complexometric Titrations

3. Estimation of $\text{Mg}^{+2}/\text{Zn}^{+2}/\text{Ca}^{+2}$ in given solution by using standard EDTA solution.

Preparation of Polymers (only demonstration)

4. Nylon Fibers
5. Bakelite

Redox Titrations

6. Estimation of Mohr's salt by using potassium permanganate.
7. Estimation of Mohr's salt by using potassium dichromate.
8. Estimation of copper (II) ion using standard hypo solution.

Conductometric Measurements

9. Estimation of amount of HCl conductometrically using standard NaOH solution.
10. Estimation of amount of HCl conductometrically using NH_4OH solution.

Potentiometric Measurements

11. Estimation of amount of HCl potentiometrically using NaOH solution.

Estimations

12. Measuring pH of the given sample solution using pH meter (demonstration only).
13. Estimation of Vitamin C in a given sample.

Colorimetric Analysis

14. Determination of Iron (III) by colorimetric method.

REFERENCES

Lab manual

B.Tech. (I Sem.)**20CS51 - PROGRAMMING FOR PROBLEM SOLVING USING C LAB**

| L | T | P | Cr. |
|----------|----------|----------|------------|
| 0 | 0 | 3 | 1.5 |

Pre-requisite : NIL

Course Educational Objective: The objective of the course is to learn the basic elements of C Programming Structures like Data Types, Expressions, Control Statements, and Various I/O Functions and to solve simple mathematical problems using control structures. Design and implementation of various software components, which solve real world problems.

Course Outcomes (CO): *At the end of this course, the student will be able to:*

CO1: Apply control structures of C in solving computational problems. (**Apply - L3**)

CO2: Implement derived data types & use modular programming in problem solving (**Apply- L3**)

CO3: Implement user defined data types and perform file operations. (**Apply- L3**)

CO 4: Improve individual / teamwork skills, communication & report writing skills with ethical values.

of modules at most 10 can be taught and all the modules should be in line with theory.

Module 1: Introduction to Raptor Tool.

Module 2: Problem solving using Raptor Tool.

Module 3: Exercise Programs on Basics of C-Program.

Module 4: Exercise Programs on Control Structures.

Module 5: Exercise Programs on Loops & nesting of Loops.

Module 6: Exercise Programs on Arrays & Strings.

Module 7: Exercise Programs on Pointers.

Module 8: Exercise Programs on Functions.

Module 9: Exercise Programs on user defined data types.

Module 10: Exercise Programs on Files.

**20ME53 - COMPUTER AIDED ENGINEERING
DRAWING**

B.Tech. (I Sem.)

| L | T | P | Cr. |
|----------|----------|----------|------------|
| 0 | 0 | 3 | 1.5 |

PRE-REQUISITES: Nil

COURSE EDUCATIONAL OBJECTIVE:

The main objectives of this course are to familiarize various commands used in Auto-CAD and to visualize the isometric and orthographic views of any solid object.

COURSE OUTCOMES:

After completion of the course students are the able to:

- CO1** Understand the Auto-CAD basics and apply to solve practical problems used in industries where the speed and accuracy can be achieved. (**Understand-L2**)
- CO2** Understand the principle of Orthographic projections of points, lines, planes and solids. (**Understand-L2**)
- CO3** Draw the isometric views of lines, planes and simple solids. (**Understand-L2**)
- CO4** Convert orthographic to isometric vice versa. (**Understand-L2**)

At least 10 Exercises are to be conducted using Auto Cad software:

BASIC AUTO CAD COMMANDS:

1. Basic drawing commands (line, circle, arc, ellipse, polygon, and rectangle).
2. Edit commands (copy, move, erase, zoom).
3. Array commands (polar array, rectangular array, P-edit, divide a line, offset).
4. Hatching & line commands (hatching with different angles& different types of lines).
5. Mirror & trim commands (mirror an object, trim, extend a line, chamfer& fillet, explode).
6. Dimensioning & text commands (linear, angular, radius, diameter& text).

PROJECTION OF POINTS AND LINES:

1. Projection of points (I, II, III, & IV quadrants).
2. Projection of lines parallel to both reference planes.
3. Projection of lines parallel to one reference plane & inclined to other reference plane.

PROJECTION OF PLANES AND SOLIDS:

1. Projection of planes parallel to one reference plane & perpendicular to other reference plane.
2. Projection of planes inclined to one reference plane& perpendicular to other reference plane.
3. Projection of solids in simple position.
4. Projection of solids with axes inclined to one reference plane & parallel to other.

CONVERSION OF ORTHOGRAPHIC PROJECTIONS INTO ISOMETRIC PROJECTIONS & VICE VERSA:

1. Conversion of plane objects.
2. Conversion of circular objects.
3. Conversion of both combination of plane figures and circular objects.

| Expt. No. | Type of Drawings | Name of the Experiment |
|------------------|-----------------------------------------------|-------------------------------------------------------------------|
| 1 | Basic drawing Commands | Exercise on Basic Drawing Commands-I |
| 2 | | Exercise on Basic Drawing Commands-II |
| 3 | Modify commands | Exercise on Modify Commands |
| 4 | Projection of Lines | Exercise on Projection of Lines-I |
| 5 | | Exercise on Projection of Lines-II |
| 6 | Isometric Diagrams | Exercise on isometric views-I |
| 7 | | Exercise on isometric views-II |
| 8 | | Exercise on isometric views-III |
| 9 | | Exercise on isometric views-IV |
| 10 | Conversion of Isometric to Orthographic views | Exercise on conversion of Isometric views into Orthographic views |

REFERENCES

1. M. Kulkarni, A.P Rastogi, and A.K. Sarkar, Engineering Graphics with AutoCAD, PHI Learning Private Limited, New Delhi, 2009.
2. Bethune, Engineering Graphics with AutoCAD, PHI Learning Private Limited, New Delhi, 2009.
3. N. D. Bhatt, Engineering Drawing, 51th Revised and Enlarged Edition, Charotar Publishers, 2012.

| L | T | P | Cr. |
|---|---|---|-----|
| 2 | 0 | 0 | 2 |

B.Tech.(IISem.)

20FE02 - PROFESSIONAL COMMUNICATION - II**Pre-requisites: Nil**

Course Educational Objective: To improve English language proficiency of the students on various aspects like vocabulary, grammar, communication skills, listening skills, Reading & Writing skills.

Course Outcomes: At the end of the course, the student will be able to

- CO1:** Produce a coherent paragraph interpreting a figure/graph/chart/table. (**Understand – L2**)
- CO2:** Comprehend the given texts thoroughly by guessing the meanings of the words contextually (**Understand – L2**).
- CO3:** Use language appropriately for describing/comparing/contrasting/giving directions & suggestions (**Remember – L1**).
- CO4:** Write formal/informal dialogues with an understanding of verbal/non-verbal features of communication. Guess meanings of the words from the context (**Understand – L2**).
- CO5:** Write well-structured essays; Reports & Résumé (**Apply – L3**).

UNIT - I

Fabric of Change- ‘H.G. Wells and the Uncertainties of Progress – Peter J. Bowler’; Reading: Studying the use of Graphic elements in texts; Grammar & Vocabulary: Quantifying Expressions; Adjectives and adverbs; Comparing and Contrasting; Degrees of Comparison
Writing: Information Transfer

UNIT - II

Tools for Life - ‘Leaves from the Mental Portfolio of a Eurasian – Sui Sin Far’; Reading: Global Comprehension; Detailed Comprehension; Grammar & Vocabulary: Active & Passive Voice; Idioms & Phrases; Writing: Structured Essays using suitable claims and evidences

UNIT - III

‘Homi Jahangir Bhabha’;

Grammar & Vocabulary: Words often confused; Common Errors; Writing: Incident & Investigation Reports.

UNIT - IV

‘Jagadish Chandra Bose’; Grammar & Vocabulary: Use of antonyms; Correction of Sentences; Writing: Dialogue Writing.

UNIT - V

‘Prafulla Chandra Ray’; Grammar & Vocabulary: Analogy; Sentence Completion; Writing: Writing a Résumé

TEXTBOOKS:

1. Prabhavati. Y & et al, “English All Round – Communication Skills for Undergraduate Learners”, Orient Black Swan, Hyderabad, 2019.
- 2 “The Great Indian Scientists” published by Cengage Learning India Pvt. Ltd., Delhi, 2017

REFERENCE BOOKS:

1. Swan,M., “Practical English Usage”, Oxford University Press, 2016.
2. Kumar,S and Latha, P, “Communication Skills”, Oxford University Press, 2018.
3. Rizvi Ashraf M., “Effective Technical Communication”, Tata Mc Graw Hill, New Delhi, 2008.
4. BaradwajKumkum, “Professional Communication”, I.K.International Publishing House Pvt.Lt., New Delhi, 2008.
5. Wood,F.T., “Remedial English Grammar”, Macmillan, 2007.

B.Tech. (II Sem.)

**20FE04 - LINEAR ALGEBRA AND
TRANSFORMATION TECHNIQUES**

| L | T | P | Cr. |
|---|---|---|-----|
| 2 | 1 | 0 | 3 |

Pre-requisites: Nil

Course Educational Objective: In this course, students learn Matrix Algebra and introduced with transformation techniques such as Laplace Transforms and Z – Transforms.

Course Outcomes: At the end of the course, the student will be able to

- CO1:** Investigate the consistency of the system of equations and solve them (**Apply – L3**).
- CO2:** Determine the eigen vectors and inverse, powers of a matrix using Cayley – Hamilton theorem (**Apply – L3**).
- CO3:** Use the concepts of Laplace Transforms to various forms of functions. (**Understand – L2**).
- CO4:** Solve ordinary differential equations by using Laplace Transforms (**Apply – L3**).
- CO5:** Apply Z - Transforms to solve difference equations(**Apply – L3**).

UNIT – I**System of Linear Equations**

Matrices - Rank- Echelon form, Normal form, PAQ form– Solution of Linear Systems – Homogeneous system of equations and Non-Homogeneous system of equations.

UNIT – II**Eigen Values and Eigen Vectors**

Eigen values – Eigen Vectors – Properties – Cayley-Hamilton Theorem – Inverse and Powers of a matrix by using Cayley-Hamilton Theorem.

UNIT – III**Laplace Transforms**

Laplace transforms of standard functions –Linear Property - Shifting Theorems, Change of Scale Property
Multiplication and Division by ‘t’ - Transforms of derivatives and integrals – Unit step function – Dirac’s delta function.

UNIT – IV**Inverse Laplace Transforms**

Inverse Laplace transforms– Linear Property - Shifting Properties - Convolution theorem, Applications of Laplace transforms to ordinary differential equations.

UNIT – V**Z-Transforms**

Z-transform – properties – Damping rule – Shifting rule – Initial and final value theorems - Inverse Z –transform - Convolution theorem – Solution of difference equation by Z-transforms.

TEXTBOOKS:

1. B.S. Grewal, “*Higher Engineering Mathematics*”, 42nd Edition, Khanna Publishers, New Delhi, 2012.
2. B. V. Ramana, “*Higher Engineering Mathematics*”, 1st Edition, TMH Publications, New Delhi, 2010.

Reference Books:

1. M. D. Greenberg, “*Advanced Engineering Mathematics*”, 2nd Edition, TMH Publications, New Delhi, 2011.
2. Erwin Kreyszig, “*Advanced Engineering Mathematics*”, 8th Edition, John Wiley & sons, New Delhi, 2011.
3. W.E. Boyce and R. C. DiPrima, “*Elementary Differential Equations*”, 7th Edition, John Wiley & sons, New Delhi, 2011.

B.Tech. (II Sem.)

20FE07 - APPLIED PHYSICS

| L | T | P | Cr. |
|---|---|---|-----|
| 2 | 1 | 0 | 3 |

Pre-requisites: Nil

Course Educational Objectives: It enables the students to understand the fundamental concepts of optics, quantum mechanics, free electron theory of metals, semiconductors, dielectrics and their applications.

Course Outcomes: At the end of this course, the student will be able to

CO1: Define the nature of interference and diffraction (**Remember – L1**).

CO2: Apply the lasers and optical fibers in different fields (**Apply – L3**).

CO3: Estimate the electrical conductivity of metals (**Understand – L2**).

CO4: Analyze the properties of semiconducting materials (**Understand – L2**).

CO5: Classify the different types of magnetic and dielectric materials (**Understand – L2**).

UNIT – I**Wave Optics**

Interference: Principle of super position, Conditions for Interference, Interference in thin parallel film by reflection, Newton's rings (reflection), working principle of Interferometer.

Diffraction: Introduction, Fraunhofer diffraction at single slit- Diffraction due to circular aperture, Diffraction Grating- Resolving power of Grating.

UNIT – II**Lasers and optical fibers**

Lasers: Introduction - Principle of laser (absorption, spontaneous and stimulated emission of radiation), Einstein Coefficients – Nd-YAG laser, Helium Neon laser- applications.

Optical Fibers: Optical Fiber principle, Structure of optical fiber, numerical aperture and acceptance angle, types of optical fibers - applications.

UNIT – III**Principles of Quantum Mechanics & Free electron theory**

Principles of quantum mechanics: de Broglie Hypothesis, Davisson - Germer experiment, Schrodinger time independent and dependent wave equations, physical significance of the wave function – particle in a box.

Free electron theory: Classical free electron theory- Postulates, Advantages and Draw backs, Fermi-Dirac distribution function-Temperature dependence of Fermi- Dirac distribution function, Classification of Solids on the basis of Band theory.

UNIT – IV**Semiconductor physics**

Conductivity of Intrinsic and Extrinsic semiconductors, Drift and Diffusion Current, Einstein relation, Hall Effect, Differences between direct and indirect Band Gap semiconductors, Solar Cell, Applications of Solar Cells.

UNIT – V:**Magnetic & Dielectric materials**

Magnetic parameters, Classification of magnetic materials-Diamagnetic, paramagnetic and ferromagnetic materials, Hysteresis loop, soft and hard magnetic materials, Applications of Ferro magnetic materials

Dielectrics: polarization - Electronic and ionic polarization, orientation polarization (Qualitative), Local field, Claussius Mosotti equation, Applications of dielectric materials.

TEXTBOOKS:

- 1.V. Rajendran, “*Engineering Physics*”, TMH, New Delhi, 6th Edition, 2014.
- 2.M.N. Avadhanulu, P.G. Kshirsagar, “*Engineering Physics*”, S. Chand & Co., 2nd Edition, 2014.

REFERENCE BOOKS:

1. M.N. Avadhanulu, TVS Arun Murthy, “*Applied Physics*”, S. Chand & Co., 2nd Edition, 2007.
2. P.K. Palani Samy, “*Applied Physics*”, Sci. Publ. Chennai, 4th Edition, 2016.
3. P. Sreenivasa Rao, K Muralidhar, “*Applied Physics*”, Him. Publi. Mumbai, 1st Edition, 2016.
4. Hitendra K Mallik , AK Singh “ *Engineering Physics*”, TMH, New Delhi, 1st Edition, 2009.

B.Tech. (II Sem.)

20CS05 - PYTHON PROGRAMMING

| L | T | P | Cr. |
|---|---|---|-----|
| 3 | 0 | 0 | 3 |

Pre-requisite : Problem Solving Skills

Course Educational Objective: The objective of the course is to provide basic knowledge of python. python programming is intended for software engineers, system analysts, program managers and user support personnel who wish to learn the Python programming language for Problem solving and programming capability.

Course Outcomes: At the end of this course, the student will be able to

- CO1:** Identify basic building blocks of python to solve mathematical problems. (**Understand-L2**)
- CO2:** Apply the in-built data structures like list, tuple, set and dictionary for solving problems. (**Apply - L3**)
- CO3:** Use exception-handling mechanism to catch run-time errors (**Apply - L3**)
- CO4:** Demonstrate compelling concepts about string manipulation, regular expressions, and file handling. (**Understand - L2**)
- CO5:** Demonstrate object-oriented programming principles of python. (**Understand - L2**)

UNIT-I

Introduction to Python: History of Python, Usage of Python interpreter, Python Shell, Indentation, Python Built-in types, Variables, Assignment, Input-Output Statements, Simple Programs, Identifiers and keywords, Literals. **Operators:** Arithmetic operators, Relational operators, Logical operators, Assignment operators, Bit-wise operators, Python Membership Operators (in & not in), Python Identity Operators (is & is not), Operator precedence.

Control Structures: Conditional Statements - if, if-else, Nested if-else. Jumping Statements - continue, break, and pass. Python Loops - while, for, Nested loops with Programming Examples, Mathematical Functions and Constants (import math), Random Number Functions.

UNIT-II

Lists: Concept, Creating and Accessing Elements, Updating & Deleting Lists, basic List Operations, Reverse, Indexing, Slicing and Matrices, Built-in List Functions.

Tuples: Introduction, Creating & Deleting Tuples, Accessing values in a Tuple, Updating tuples, Delete Tuple Elements, basic Tuple Operations, Indexing, Slicing and Matrices, built- in tuple Functions.

Sets: Concept, Operations.

Dictionaries: Introduction, Accessing values in dictionaries, working with dictionaries, Properties, Functions.

UNIT-III

Functions: Defining a Function, Calling a Function, Types of Functions, Function Arguments, Anonymous functions, Global and Local Variables, Recursion

Modular Design: Creating modules, import statement, from, Date and Time Module.

Exception Handling: Exception, Exception Handling, except clause, Try, finally clause, User Defined Exceptions.

UNIT-IV

Python strings: Concept, Slicing, Escape characters, String Special Operations, String formatting Operator, Triple Quotes, Raw String, Unicode Strings, and Built-in String methods.

Regular Expression Operations: Using Special Characters, Regular Expression Methods, Named Groups in Python Regular Expressions, Regular Expression with glob Module.

Files: Introduction to files, file operations- Read, write, and search

UNIT-V

Object Oriented Programming OOP in Python: Classes, 'self-variable', Methods, Constructor Method, Inheritance, Overriding Methods, and Data hiding.

TEXTBOOKS:

1. Reema Thareja, “Python Programming Using Problem Solving Approach”, Oxford Publications
2. Python for Everybody: Exploring Data In Python 3 by [Dr. Charles Russell Severance](#), [Sue Blumentberg](#)

Reference books:

1. Gowrishankar S and Veena A, “Introduction to Python Programming”, CRC Press, Taylor, and Francis Group – A CHAPMAN & HALLBOOK.
2. R. Nageswara Rao, “Core python programming”, Dreamtech, 2017.
3. Y. Daniel Liang, “Revel for Introduction to Python Programming and Data Structures”, Pearson Publications.

B.Tech. (II Sem.)

20CS03 - DATA STRUCTURES

| L | T | P | Cr. |
|---|---|---|-----|
| 3 | 0 | 0 | 3 |

Pre-requisite : Programming Language

Course Educational Objectives:

The objective of the course is to make students familiar with writing algorithms to implement different data structures like stacks, queues, trees and graphs, and various sorting techniques.

Course Outcomes (COs): *At the end of this course, the student will be able to*

- CO 1: Write the algorithms for various operations on list using arrays and linked list and analyze the time complexity of its operations.(**Understand - L2**)
- CO 2: Apply linear data structures like stack and queue in problem solving.(**Apply - L3**)
- CO 3: Demonstrate various searching and sorting techniques and compare their computational complexities in terms of space and time.(**Understand - L2**)
- CO 4: Write the algorithms for various operations on binary trees, binary search trees and AVL trees.(**Understand - L2**)
- CO 5: Demonstrate graph traversal techniques and hashing techniques.(**Understand - L2**)

UNIT - I

Algorithm Analysis:

Introduction to Algorithm, Algorithm Analysis, Asymptotic Notations.

Introduction to arrays and Abstract Data Type (ADT)

Lists: List using arrays and linked list- Singly Linked List, Doubly Linked List, Circular LinkedList.

UNIT – II

Stacks: Stack ADT, Implementation using arrays and linked list.

Applications of stacks: Infix to postfix expression conversion, Evaluation of Postfix expressions and balancing the symbols.

Queues:

Queue: Queue ADT, Implementation of Queue using arrays and linked list, circular queue, DEQUE

UNIT - III

Sorting: Bubble sort, Insertion Sort, Selection sort, Merge Sort, Quick Sort & Heap Sort

UNIT - IV

Trees: Introduction, Tree traversals, Binary Trees, Binary Search Trees, Balanced Binary search tree - AVL Trees and its operations.

UNIT - V

Graphs: Fundamentals, Representation of graphs, Graph Traversals: BFS, DFS.

Hashing: Hash Table, Hash Function, Collision resolution Techniques- separate Chaining,

Open addressing, rehashing.

TEXTBOOKS:

1. Mark Allen Weiss, "Data Structures and Algorithm Analysis in C", Pearson Education, 2nd edition [1,2,3 units].
2. ReemaThareja, Data Structures using c, Oxford Publications [3,4,5].

REFERENCES:

1. Langson, Augenstein & Tenenbaum, 'Data Structures using C and C++', 2nd Ed, PHI.
2. RobertL.Kruse, Leung and Tando, 'Data Structures and Program Design in C', 2nd edition, PHI.

B.Tech. (II Sem.)

20MC01 - CONSTITUTION OF INDIA

| L | T | P | Cr. |
|---|---|---|-----|
| 2 | 0 | 0 | 0 |

Pre-requisites: Nil**Course Educational Objectives**

- To enable the student to understand the importance of constitution.
- To understand the structure of Executive, Legislature and Judiciary.
- To understand Philosophy of fundamental rights and duties.
- To understand the autonomous nature of constitution bodies like Supreme Court and High Court Controller and Auditor General of India and Election Commission of India.
- To understand the Central and State relation, financial and administrative.

Course Outcomes: *At the end of the course, the student shall be able to*

- CO 1: Understand history and philosophy of constitution with reference to Preamble, Fundamental Rights and Duties (**Understand – L2**).
- CO 2: Understand the concept of Unitary and Federal Government along with the role of President, Prime Minister and Judicial System (**Understand – L2**).
- CO 3: Understand the structure of the state government, Secretariat, Governor and Chief Minister and their functions (**Understand – L2**).
- CO 4: learn local administration viz. Panchayat, Block, Municipality and Corporation (**Understand – L2**)..
- CO 5: learn about Election Commission and the process and about SC, ST, OBC and women (**Understand – L2**).

UNIT – I:

Introduction to Indian Constitution: ‘Constitution’ meaning of the term, Indian Constitution – Sources and Constitutional History, Features – Citizenship, Preamble, Fundamental Rights and Duties, Directive Principles of State Policy.

UNIT – II:

Union Government and its Administration Structure of the Indian Union: Federalism Centre – State relationship, President: Role, Power and Position. Prime Minister (PM) and Council of Ministers, Cabinet and Central Secretariat, Lok Sabha, Rajya Sabha. The Supreme Court and High Court: Powers and Functions.

UNIT – III:

State Government and its Administration Governor – Role and Position – Chief Minister (CM) and Council of Ministers. State Secretariat: Organization, Structure and Functions.

UNIT – IV:

A Local Administration -- Role and Importance, Municipalities – Mayor and Role of Elected Representative, Panchayati Raj: Functions of Panchayati Raj Institution, Zilla Panchayat, Elected Officials and their roles, Village level – Role of Elected and Appointed officials.

UNIT – V:

Election Commission: Election Commission – Role of Chief Election Commissioner and Election Commissionerate State Election Commission: Functions and Commissions for the welfare of SC/ST/OBC and Women.

Reference Books

1. Durga Das Basu, Introduction to the Constitution of India, Prentice – Hall of India Pvt.Ltd., New Delhi.
2. Subash Kashyap, Indian Constitution, National Book Trust.
3. J.A. Siwach, Dynamics of Indian Government and Politics.
4. D.C. Gupta, Indian Government and Politics.
5. H.M.Sreevai. Constitutional Law of India, 4th edition in 3 volumes (Universal LawPublication).
6. J.C. Johari, Indian Government and Politics Hans.
7. J.Raj, Indian Government and Politics.
8. M.V. Pylee, Indian Constitution, Durga Das Basu, Human Rights inConstitutional Law, Prentice – Hall of India Pvt. Ltd., New Delhi.
9. Noorani, A.G. (South Asia Human Rights Documentation Centre), Challenges to Civil Right). Challenges to Civil Rights Guarantees in India, Oxford University Press 2012.

E-Resources:

1. nptel.ac.in/courses/109104074/8.
2. nptel.ac.in/courses/109104045.
3. nptel.ac.in/courses/101104065.
4. www.hss.iitb.ac.in/en/lecture-details.
5. www.iitb.ac.in/en/event/2nd-lecture-institute-lecture-series-indianconstitution

B.Tech. (II Sem.)

20FE54 - APPLIED PHYSICS LAB

| L | T | P | Cr. |
|---|---|---|-----|
| 0 | 0 | 3 | 1.5 |

Pre-requisites : Nil

Course Educational Objective: This course enables the students to acquire theoretical ideas, analytical techniques, and graphical analysis, by completing a host of experiments with the procedures and observational skills for appropriate use of simple and complex apparatus.

COURSE OUTCOMES: At the end of the course, the student will be able to:

- CO1:** Analyze the wave characteristics of light (**Understand – L2**).
- CO2:** Estimate the magnetic field using Stewart's and Gee's apparatus (**Understand – L2**).
- CO3:** Verify the characteristics of semiconductor diodes (**Apply – L3**).
- CO4:** Determine the acceptance angle and numerical aperture of optical fiber (**Apply – L3**).
- CO5:** Improve report writing skills and individual teamwork with ethical values (**Understand – L2**).

List of Experiments

(Any of the 10 experiments are required to be conducted)

GENERAL EXPERIMENTS:

1. Determine the energy band gap of a semiconductor Diode.
2. Study the characteristics of Zener Diode.
3. Study the magnetic field along the axis of a current carrying circular coil using Stewart's & Gee's apparatus and to verify Biot - Savart's law.
4. Study the characteristics of Solar cell
5. Determination of dielectric constant by charging and discharging method.
6. Study the characteristics of Photo diode.
7. Determination of resistivity of semiconductor by four probe method.

OPTICS LAB EXPERIMENTS:

8. Determine the wavelength of a laser radiation.
9. Determine the width of a single slit by forming diffraction pattern.
10. Determine the Radius of Curvature of a Plano - Convex lens by forming Newton's Rings.
11. Determine the Wavelengths of various spectral lines by using diffraction grating.
12. Resolving power of grating.
13. Determine the acceptance angle and numerical aperture of a fiber.
14. Measure the bending losses in the optical fiber cable at different wavelengths.

B.Tech. (II Sem.)

20CS54 - PYTHON PROGRAMMING LAB

| L | T | P | Cr. |
|---|---|---|-----|
| 0 | 0 | 3 | 1.5 |

Pre-requisite: Basic Knowledge of Programming.

Course Educational Objective:

The Objective of Python course is to lead the students from the basics of writing and running Python scripts in problem solving and to design and implement the modules and understands the working of classes and objects in python.

Course Outcomes: *At the end of the course, the student shall be able to*

- CO 1:** Apply building blocks of Python in solving computational problems. (**Apply - L3**)
- CO 2:** Implement in-built data structures available in Python to solve computational problems. (**Apply - L3**)
- CO 3:** Implement modular programming, string manipulations and Object-oriented programming in python. (**Apply - L3**)
- CO 4:** Improve individual / teamwork skills, communication & report writing skills with ethical values.

Introduction: Language basics and example problems (Two weeks)

- a) Implement Python Script for checking the given year is leap year or not.
- b) Implement Python Script for finding biggest number among 3 numbers.
- c) Implement Python Script for displaying reversal of a number.
- d) Implement Python Script to check given number is Armstrong or not.
- e) Implement Python Script to print sum of N natural numbers.
- f) Implement Python Script to check given number is palindrome or not.
- g) Implement Python script to print factorial of a number.
- h) Implement Python Script to print all prime numbers within the given range.
- i) Implement Python Script to calculate the series: $S=1+x+x^2+x^3+\dots+x^n$
- j) Implement Python Script to print the following pattern:

```

      *
     * *
    * * *
  
```

Module 1: Exercise Programs on Lists.

- a) Write a Python script to display elements of list in reverse order.

- b) Write a Python script to find the minimum and maximum elements without using built-in operations in the lists.
- c) Write a Python script to remove duplicates from a list.
- d) Write a Python script to append a list to the second list.
- e) Write a Python script to count the number of strings in a list where the string length is 2 or more.

Module 2: Exercise Programs on Tuples.

- a) Write a Python script to create a tuple with different data types.
- b) Write a Python script to find the repeated items of a tuple.
- c) Write a Python script to replace last value of tuples in a list.
Sample list: [(10, 20, 40), (40, 50, 60), (70, 80, 90)]
Expected Output: [(10, 20, 100), (40, 50, 100), (70, 80, 100)]
- d) Write a Python script to sort a tuple by its float element.
Sample data: [('item1', '12.20'), ('item2', '15.10'), ('item3', '24.5')]
Expected Output: [('item3', '24.5'), ('item2', '15.10'), ('item1', '12.20')]

Module 3: Exercise Programs on Sets and Dictionaries.

- a) Write a Python script to add member(s) in a set.
- b) Write a Python script to perform Union, Intersection, difference and symmetric difference of given two sets.
- c) Write Python script to test whether every element in S is in T and every element in T is in S.
- d) Write a Python script to sort (ascending and descending) a dictionary by value.
- e) Write a Python script to check whether a given key already exists or not in a dictionary.
- f) Write a Python script to concatenate following dictionaries to create a new one.
- g) Sample Dictionary : dic1={1:10, 2:20} dic2={3:30, 4:40} dic3={5:50,6:60}
Expected Result : {1: 10, 2: 20, 3: 30, 4: 40, 5: 50, 6: 60}
- h) Write a Python script to print a dictionary where the keys are numbers between 1 and 15 (both included) and the values are square of keys.
- i) Write a Python program to map two lists into a dictionary.

Module 4: Exercise Programs on functions and recursion.

- a) Define a function max_of_three() that takes three numbers as arguments and returns the largest of them.
- b) Write a program which makes use of function to display all such numbers which are divisible by 7 but are not a multiple of 5, between given range X and Y.

- c) Define functions to find mean, median, mode for the given numbers in a list.
- d) Define a function which generates Fibonacci series up to n numbers.
- e) Implement a python script for factorial of number by using recursion.
- f) Implement a python script to find GCD of given two numbers using recursion.

Module 5: Exercise programs on Date and Time Modules.

- a) Write a Python script to get the current time in Python.
- b) Write a Python script to get current time in milliseconds in Python
- c) Write a Python script to print next 5 days starting from today.

Module 6: Exercise programs on Exception Handling.

- a) Write a Python script to handle simple errors by using exception handling mechanism.
- b) Write a Python script to handle multiple errors with one except statement.

Module 7: Exercise programs on Strings

- a) Implement Python Script to perform various operations on string using string libraries.
- b) Implement Python Script to check given string is palindrome or not.
- c) Implement python script to accept line of text and find the number of characters, number of vowels and number of blank spaces in it.
- d) Implement python script that takes a list of words and returns the length of the longest one.

Module 8: Exercise programs on Regular Expressions

- a) Write a Python script to check that a string contains only a certain set of characters (in this case a-z, A-Z and 0-9).
- b) Write a Python script to check whether password is valid or not.

Conditions for a valid password are:

Should have at least one number.

Should have at least one uppercase and one lowercase character.

Should have at least one special symbol.

Should be between 6 to 20 characters long.

Module 9: Exercise programs on Object Oriented Programming

- a) Write a Python script to create and access class variables and methods.
- b) Write a Python script to implement method overloading.
- c) Write a Python script to implement single inheritance.
- d) Write a Python script to implement method overriding.

Module 10: Exercise programs on Python Libraries – Numpy , Pandas , Matplotlib

- a) Write a NumPy program to generate a matrix product of two arrays.
- b) Write a NumPy program to create a random array with 1000 elements and compute the average, variance, standard deviation of the array elements.
- c) Demonstrate how to download dataset and how to create DataFrame
 - i. Write a Pandas program to get the first 3 rows of a DataFrame
 - ii. Write a Pandas program to select the specified columns and rows from a given data frame.
 - iii. Write a Pandas program to select the rows where the score is missing, i.e. is NaN.
 - iv. Write a Pandas program to insert a new column in existing DataFrame.
- d) Write a Python programming to display a bar chart using different color for each bar.
- e) Write a Python programming to create a pie chart with a title.

B.Tech. (II Sem.)

20CS53 - DATA STRUCTURES LAB

| L | T | P | Cr. |
|---|---|---|-----|
| 0 | 0 | 3 | 1.5 |

Pre-requisite : Programming Language

Course Educational Objectives:

The objective of this course is to make students familiar with writing algorithms to implement different data structures like stacks, queues, trees and graphs, and various sorting techniques.

Course Outcomes (COs): *At the end of this course, the student will be able to*

- CO 1: Implement Linear Data Structures using array and Linked list. (**Apply - L3**)
- CO 2: Implement Various Sorting Techniques. (**Apply - L3**)
- CO 3: Implement Non-Linear Data Structure such as Trees & Graphs. (**Apply - L3**)
- CO 4: Improve individual / teamwork skills, communication & report writing skills with ethical values.

I) Exercise Programs on List ADT

- a) Implementation of List using Arrays.
- b) Implementation of List using Linked List.

II) Exercise Programs on Stacks & Queue ADT

- a) Implementation of Stack Operations using Arrays.
- b) Implementation of Stack Operations using Linked List.
- c) Implementation of Queue Operations using Arrays.
- d) Implementation of Queue Operations using Linked List.

III) Exercise Programs on Stack Applications

- a) Conversion of Infix Expression to postfix Expression.
- b) Conversion of Infix Expression to prefix Expression.
- c) Evaluation of Postfix Expression
- d) Implementation of Balancing Symbols.

IV) Exercise Programs on Types of Queues

- a) Implementation of Circular Queues Linked List.

b) Implementation of Double Ended Queue using Arrays.

c) Implementation of Double Ended Queue using Linked List.

V) Exercise Programs on Sorting Techniques.

a) Implementation of Insertion Sort and

b) Implementation of Selection Sort.

c) Implementation of Merge Sort.

d) Implementation of Quick Sort.

e) Implementation of Bubble Sort.

f) Implementation of Heap Sort.

VI) Exercise Programs on Trees

a) Implementation of Binary Tree Traversals.

b) Implementation of Binary Search Tree Operations.

VII) Exercise Programs on Graph Traversal Techniques.

a) Breadth First Search (BFS)

b) Depth First Search (DFS)

B.Tech.(II Sem.)

20IT52 - MATHEMATICAL APPLICATIONS LAB

| L | T | P | Cr. |
|---|---|---|-----|
| 0 | 0 | 2 | 1 |

Pre-requisites: Basic Mathematics

Course Educational Objective: The Students will be able to learn the basic usage of MATLAB/SCI Lab or some other open-source tools to solve basic mathematical problems.

Course Outcomes: At the end of the course, the student will be able to:

CO1: Identify basic elements of programming structures. (**Understand – L2**)

CO2: Implement elementary mathematical operations using MATLAB/ SCI Lab or some other open-source tools. (**Apply-L3**)

CO3: Implement the binary operations using MATLAB/ SCI Lab or some other open-source tools. (**Apply-L3**)

CO 4: Improve individual / teamwork skills, communication & report writing skills with ethical values.

List of Experiments

(Any of the 10 experiments are required to be conducted)

1. To know the history and features of MATLAB.
2. To know the local environment and basic commands of MATLAB.
3. To know the history and features of SCI LAB.
4. To know the local environment and basic commands of SCI LAB.
5. Perform elementary mathematical operations like addition, subtraction, multiplication, and division.
6. Perform elementary logical operations.
7. Perform basic operations on matrices (like addition, subtraction, multiplication) and display specific rows or columns of the matrix.
8. Perform Incrementing and Decrementing operation.
9. Perform elementary Bitwise operations.
10. Write a program to implement Array arithmetic operations.
11. Write a program to implement Multidimensional view of data.
12. Implement the basic statistical operations with the help of MATLAB / SCILAB.
13. Implement the Trigonometric Functions with the help of MATLAB / SCILAB.
14. Implement the conditional statements with the help of MATLAB/ SCILAB.

15. Evaluate the expression $a^3 + \sqrt{bd} - 4c$ where a=1.2, b=2.3, c=4.5 and d=4.

B.Tech. (III Sem.)

20FE09 - PROBABILITY AND STATISTICS

| L | T | P | Cr. |
|---|---|---|-----|
| 3 | 0 | 0 | 3 |

Pre-requisite : Nil

Course Educational Objective: The objective of this course is to provide students with the foundations and applications of probabilistic and statistical methods mainly used in varied applications in engineering and science.

Course Outcomes: At the end of this course, the student will be able to

- CO1:** Understand various probabilistic situations using the various laws of probability and random variables (**Understand - L2**)
- CO2:** Apply probability distributions like Binomial, Poisson, Normal and Exponential distributions in solving engineering problems (**Apply - L3**)
- CO3:** Calculate the standard error of sampling distribution and confidence intervals for parameters like mean and proportion based on the sample data. (**Apply - L3**)
- CO4:** Analyze the data scientifically with the appropriate statistical methodologies to apply the suitable test of hypothesis (**Analyze - L4**)
- CO5:** Construct the regression lines to predict the dependent variables and calculate the Correlation Coefficient for a bivariate statistical data. (**Apply - L3**)

UNIT – I

Probability and Random variables

Probability, Sample space and events, Additive Rule, Conditional probability, Multiplicative rule, Baye's theorem.

Random variables – Discrete and continuous Random Variables, distribution function. Mathematical Expectation of one-dimensional Random Variable.

UNIT – II

Probability Distributions

Binomial distribution , Poisson distribution , Poisson approximation to Binomial distribution, Exponential distribution, Normal distribution , Normal approximation to Binomial distribution

UNIT – III

Sampling distribution & Estimation

Population, sample, parameter, statistic, sampling distribution, Standard error, Types of sampling, Sampling distribution of means and sampling distribution of variance, Parameter estimations –point estimation and interval estimation for mean and proportions.

UNIT – IV

Tests of Hypothesis

Hypothesis, Null and Alternate Hypothesis, , Type I and Type II errors, level of significance. Z-test for means and proportions, t-test for single mean, difference of means, paired t-test, F-test for equality of population variances, χ^2 - test for goodness of fit and independence of attributes.

UNIT – V

Correlation & Regression

Karl Pearson's coefficient of correlation, linear Regression, Regression lines, Regression coefficients, Spearman's Rank correlation coefficient, Spearman's Rank correlation for repeated ranks.

TEXTBOOKS:

1. Jay L.Devore “Probability and Statistics for engineering and the sciences.” , 8th edition, Cengage Learning india, 2012.
2. S.C.Gupta, V.K.Kapoor, “Fundamentals of Mathematical Statistics”, 11thEdition, Sultan Chand and sons, New Delhi,2014

REFERENCE BOOKS:

1. Miller & Freund’s “Probability and Statistics for Engineers”,8th edition. PHI, New Delhi,2011.
2. B.V. Ramana, “Higher Engineering Mathematics”, 1st Edition, TMH, New Delhi, 2010

B.Tech. (III Sem.)

**20CS04 - DISCRETE MATHEMATICAL
STRUCTURES**

| L | T | P | Cr. |
|---|---|---|-----|
| 3 | 0 | 0 | 3 |

Pre-requisite : Basic mathematical knowledge

Course Educational Objective: The objective of the course is to perform the operations associated with relations and functions. Relate practical examples to the functions and relations and interpret the associated operations and terminology used in the context. Use formal logic proofs and/or informal but rigorous logical reasoning to, for example, predict the behavior of software or to solve problems such as puzzles.

Course Outcomes: At the end of this course, the student will be able to

- CO1:** Construct mathematical arguments using logical connectives and quantifiers to verify them. **(Apply -L3)**
- CO2:** Demonstrate the basic terminology of functions, relations, lattices and their operations. **(Understand - L2)**
- CO3:** Apply the properties of graphs to solve the graph theory problems in Computer science. **(Apply- L3)**
- CO4:** Illustrate the basic principles/techniques to solve different algebraic structures & combinatorial problems. **(Understand- L2)**
- CO5:** Solve linear recurrence relations by recognizing homogeneity using constant coefficients, characteristic roots and Generating functions. **(Apply – L3)**

UNIT – I: Mathematical Logic

Propositional logic and Predicate Calculus: Statements and Notations, Connectives, Truth Tables, Tautologies, Equivalence of Formulas, Tautological Implications, Normal Forms, Theory of Inference for Statement Calculus, Consistency of Premises, Indirect Method of Proof, Predicative Logic, Statement Functions, Variables and Quantifiers.

UNIT – II: Sets, Relations & Functions

Introduction to Sets, representation of Sets, Operation on Sets, Properties of Binary Relations, Relation Matrix, Operations on Relations, Transitive Closure, Equivalence Relation, Compatibility and Partial Ordering Relations, Hasse Diagrams, Lattices: LUB, GLB.

Functions: Bijective Functions, Composition of Functions, Inverse Functions.

UNIT – III: Graph Theory I & II

Basic Concepts of Graphs, Matrix Representation of Graphs: Adjacency Matrices, Incidence Matrices, Isomorphic Graphs, Eulerian and Hamiltonian Graphs.

Graph Theory II: Planar Graphs, Euler's Formula, Graph Coloring, Chromatic Number, Graph Traversals: BFS, DFS. Trees: Spanning Trees: Properties, Algorithms for Minimum cost Spanning Trees

UNIT – IV: Algebraic Structures & Combinatorics

Algebraic Systems with one Binary Operation, Properties of Binary operations, Semi groups and Monoids: Homomorphism of Semi groups and Monoids, Group, Abelian group, Sub Groups, Lagrange's Theorem

Combinatorics: Basic of Counting, Permutations, Combinations, Combinations with repetition Pigeonhole Principle and its Applications, Principle of inclusion-exclusion.

UNIT – V: Recurrence Relation

Generating Function of Sequences, Calculating Coefficient of Generating Functions, Recurrence Relations, solving linear homogeneous recurrence Relations by substitution, generating functions and The Method of Characteristic Roots.

TEXTBOOKS:

1. Tremblay, Manohar, “Discrete Mathematical Structures with Applications to ComputerScience”, TMH Publications, 2008

REFERENCE BOOKS:

1. Chandrasekaran,Umaparvathi,DiscreteMathematics,PHI, 2010.
2. Ralph. P.Grimaldi, Ramana, Discrete and Combinational Mathematics,Pearson,5th edition.
3. <https://nptel.ac.in/courses/106/106/106106183/>

B.Tech. (III Sem.)

20CS07 - DATABASE MANAGEMENT SYSTEMS

| L | T | P | Cr. |
|---|---|---|-----|
| 3 | 0 | 0 | 3 |

Pre-requisite : Data Structures

Course Educational Objective: The Objective of this course is to know about basic concepts of DBMS, Database Languages, Database Design, Normalization Process, Transaction Processing, Indexing, and Interfacing with NOSQL using MongoDB.

Course Outcomes: At the end of this course, the student will be able to

- CO1:** State the Basic Components of Database Management System and data modelling using Entity-Relationship Diagrams. (**Understand -L2**)
- CO2:** Examine the relational model using Structured Query Language (SQL). (**Apply- L3**)
- CO3:** Employ principles of normalization for effective database design.(**Apply- L3**)
- CO4:** Demonstrate the necessity of transaction processing, Concurrency control mechanisms and recovery strategies in DBMS.(**Understand- L2**)
- CO5:** Describe file organization, indexing techniques and the competency in selecting NoSQL Database.(**Understand- L2**)

UNIT – I

Introduction: An overview of Database Management System, Database System Vs File System, Database System Concepts and Three Schema Architecture, Data Models, Database Schema and Instances, Data Independence, Database Languages, Database Structure.

Data Modelling using the Entity Relationship Model: ER model concepts, Notation for ER Diagram, Mapping Constraints, Keys, Concepts of Super Key, Candidate Key, Primary Key, Generalization, Aggregation, Reduction of an ER Diagrams to Tables, Relationships of Higher Degree.

UNIT – II

Relational Data Model and Language: Relational Data Model Concepts, Integrity Constraints: Entity Integrity, Referential Integrity, Key Constraints, Domain Constraints, and Relational Algebra.

Introduction to SQL: Characteristics of SQL, Advantage of SQL. SQL Data types and Literals, Insert, Update and Delete Operations, Tables, Views and Indexes, Nested Queries, Aggregate Functions, Joins, Unions, Intersection, Minus, Cursors in SQL, Triggers in SQL.

UNIT – III

Normalization: Functional Dependencies, Normal Forms - First, Second, Third Normal Forms, BCNF, Inclusion Dependencies, Loss Less Join Decompositions, Multi Valued Dependencies, Fourth Normal Form, Join Dependencies and Fifth Normal Form.

UNIT – IV

Transaction Processing Concepts: Transaction System, Testing of Serializability, Serializability of Schedules, Conflict & View Serializability, Recoverability, Deadlock Handling.

Concurrency Control Techniques: Concurrency Control, Locking Techniques for Concurrency Control, Time Stamping Protocols for Concurrency Control, Validation Based Protocol, Multiple Granularity, Recovery with Concurrent Transactions.

Crash Recovery: Log Based Recovery, Checkpoints, ARIES Algorithm

UNIT – V

Physical Database Design: Storage and file structure, indexed files, hashed files, B+ trees, files with dense index; files with variable length records.

Interfacing And Interacting With NoSQL: Introduction to NoSQL, Storing and Accessing Data, Storing Data In and Accessing Data from MongoDB, Querying MongoDB.

TEXTBOOKS:

1. Henry F. Korth, Abraham Silberschatz, S.Sudarshan, “Database System Concepts”, McGrawHill, 6th edition, 2009.
2. Shashank Tiwari, “ ProfessionalNoSql”, John Wiley & Sons, 2011.

REFERENCE BOOKS:

1. Raghuram Ramakrishnan, Johannes Gehrke, —Database Management Systems, McGrawHill, 3rd edition, 2000.
2. Date C J, —An Introduction to Database System, Pearson Education, 8th edition, 2003.
3. Ramez Elmasri, Shamkant B. Navathe, “Fundamentals of Database Systems”, Addison Wesley, 6th edition, 2010.

B.Tech. (III Sem.)

20CS08 - COMPUTER ORGANIZATION

| L | T | P | Cr. |
|---|---|---|-----|
| 3 | 0 | 0 | 3 |

Pre-requisite : IT Workshop

Course Educational Objective: The objective of the course is to learn about the functional blocks and data representation of computer system, and understands the design principles of processor & its organization and the management of memory and peripheral devices.

Course Outcomes: At the end of this course, the student will be able to

- CO1:** Evaluate digital number systems and use Boolean algebra theorems, Properties and Canonical forms for digital logic circuit design. (**Understand- L2**)
- CO2:** Design Combinational logic circuits and Sequential logic circuits. (**Apply- L3**)
- CO3:** Understand computer architecture and Data representation to perform computer arithmetic operations. (**Understand- L2**)
- CO4:** Illustrate the design principles of control unit and pipelining. (**Understand- L2**)
- CO5:** Analyze the memory hierarchy in a computer system. (**Understand- L2**)

UNIT-I

Number Systems, logic Gates and Boolean algebra

Number Systems: Binary, Octal, Decimal, Hexadecimal Number Systems. Conversion of Numbers from One, Radix to another Radix, r 's Complement and $(r-1)$'s Complements, Binary Codes, Basic Gates: NOT, AND, OR, Universal Gates: NAND, NOR, Special Gates: Ex-OR and Ex-NOR Gates

Boolean algebra

Fundamental postulates of Boolean algebra, Basic theorems and properties, Complement and Dual of Logical Expressions, SOP, POS, Minimizations of Logic Functions Using Boolean Theorems, Karnaugh Map Method (K-Map): Minimization of Boolean Functions maximum up to Four Variables, Simplifications with Don't Care Conditions Using K-Map.

UNIT-II

Combinational Logic Circuits & Sequential Logic Circuits

Design of Half Adder, Full Adder, Half Subtractor, Full Subtractor, Ripple Carry Adder. Design of Decoders, Encoders, Multiplexers, De-multiplexers, Priority Encoder,

Sequential Logic Circuits:

Latch and Flip-Flop, RS- Latch Using NAND and NOR Gates, RS, JK, T and D Flip-flops, Truth and Excitation Tables, Conversion of Flip Flops, Master-Slave Flip-flops. Registers and Counters.

UNIT-III

Functional Blocks of a Computer: CPU, Memory, Input-Output Subsystems, Control Unit. Instruction Set Architecture of A CPU-Registers, Instruction Execution Cycle, RTL Interpretation of Instructions, Addressing Modes.

Data Representation: Fixed and Floating-Point Representations. Computer Arithmetic – Integer Addition and Subtraction, Multiplication – Shift-And Add, Booth Multiplier.

UNIT-IV

CPU control unit design: Hardwired and micro-programmed design approaches. Pipelining: Basic concepts of pipelining, throughput and speedup, pipeline hazards.

UNIT-V

Memory system design: Memory hierarchy, Cache Memory and Main Memory.

Peripheral devices and their characteristics: Input-output subsystems, I/O device interface, I/O transfers–program controlled, interrupt driven and DMA.

TEXTBOOKS:

1. Morris Mano, Michael D Ciletti ,Digital Design, 4/e, 2008 , PEA.
2. CarlHamacher, ZvonksVranesic, SafeaZaky, “Computer Organization”, TMH publications

REFERENCE BOOKS:

1. M.Morris Mano, “Computer Systems Architecture”, Pearson Education publishers.[units-1,2]
2. Leach, Malvino, Saha, “Digital Logic Design”, TMH,2006
3. R.P. Jain, “Modern Digital Electronics”, TMH ,2011
4. A. Anand Kumar, “Switching Theory and Logic Design”, Prentice-Hall Of India Pvt. Limited, 2010.
5. Kohavi, Jha, Cambridge, ”Switching and Finite Automata Theory”, 3/e
6. William Stallings, “Computer Organization and Architecture”, Pearson/PHI publishers, Sixth Edition, 2015.

B.Tech. (III Sem.)

20CS09 -OBJECT ORIENTED PROGRAMMING

| L | T | P | Cr. |
|---|---|---|-----|
| 3 | 0 | 0 | 3 |

Pre-requisite : Programming for Problem Solving using C.

Course Educational Objective: The objective of the course is to learn the constructs of the Java programming language along with built-in facilities to create different applications such as console & graphical user interfaces. In the process of learning the language, they will be applying knowledge of object-oriented programming; they will get the fundamental knowledge reason collection framework.

Course Outcomes: At the end of this course, the student will be able to

- CO 1 Demonstrate the fundamentals of object-oriented programming and basic building blocks of Java. (**Understand- L2**)
- CO 2 Apply object-oriented programming principles to solve problems. (**Apply - L3**)
- CO 3 Demonstrate JAVA built-in API packages and create user-defined packages and interfaces (**Understand- L2**)
- CO 4 Develop multitasking applications using JAVA multithreaded programming and handling runtime errors using Exception Handling. (**Apply – L3**)
- CO 5 Develop GUI applications using AWT (Abstract Window Toolkit). (**Apply- L3**)

UNIT-I

Introduction to OOP: Programming paradigms, procedural programming language versus object-oriented language, principles of OOP.

Introduction to JAVA : Data types, variables, keywords, operators, and control statements.

Introduction to Classes and Object: Class definition, variables, and methods. Declaring Objects , Constructors, and this keyword.

UNIT-2

Classes and objects: overloading methods and constructors, parameter passing, returning objects, recursion. Access control, nested and inner classes, final and static keyword, variable and command-line arguments.

Inheritance and polymorphism: Inheritance, types of inheritance, super keyword, polymorphism (overloading & overriding), dynamic method dispatch, abstract class, using final with inheritance.

String handling classes: String, StringBuffer, StringTokenizer.

UNIT-3

Interfaces and packages: Interface methods, inheritance in interfaces. API : The built-in JAVA packages and creating and managing user defined packages, importance of CLASSPATH.

Exception Handling: Exception hierarchy, importance of try, catch, throw, throws and finally. Block creation of user-defined exceptions, Assertions.

UNIT-4

Multithreading: Introduction, thread life cycle, creation of threads, naming a thread, joining a thread, thread priorities, daemon thread, thread pool, thread group, thread synchronization, Inter-thread communication.

Collection Framework: Introduction, generics, collection framework hierarchy, list, set, queue, and map.

UNIT-5

AWT: AWT hierarchy, components and containers, Button, Label, Text Field, Checkbox, Choice, List, Canvas, Scrollbar, Menu Item & Menu, Container class, Layout managers (Border Layout, Flow Layout, Grid Layout, Card Layout).

Event handling: Event delegation model, event classes (Action Event, Mouse Event, Key Event, Window Event), listener interfaces (Action Listener, Mouse Listener and MouseMotionListener, Key Listener, Window Listener), adapter classes, close AWT window.

TEXTBOOKS:

1. Herbert Schildt, "Java: The complete reference", TMH Publications, 7th edition, 2006.
2. Cay S. Horstmann, "Core Java Volume I – Fundamentals", Pearson, Eleventh edition, 2018.

REFERENCE BOOKS:

1. Dr.R.NageswaraRao, "Core JAVA: An Integrated Approach", Dreamtech Press, 1st Edition 2008.
2. E. Balaguruswamy, "Programming with JAVA", TMH Publications, 2nd Edition, 2000.
3. Patrick Niemeyer & Jonathan Knudsen, "Learning Java", O'REILLY Publications, 3rd Edition, 2005.
4. Benjamin J Evans & David Flanagan, "Java-in a Nutshell – A desktop quick reference", O'REILLY Publications, 6th Edition, 2014.

B.Tech. (III Sem.)

**20CS56 - DATABASE MANAGEMENT SYSTEMS
LAB**

| L | T | P | Cr. |
|---|---|---|-----|
| 0 | 0 | 3 | 1.5 |

Pre-requisite : Programming language, Discrete Mathematical Structures, and Data Structures.

Course Educational Objective: The objective of this lab is to provide a strong formal foundation in database concepts, technology, and practice to the participants to groom them into well-informed database application developers.

Course Outcomes (CO): *At the end of this course, the student will be able to:*

- CO1:** Create & manipulate the relational database using SQL.(**Apply- L3**)
- CO2:** Implement Views, procedures, triggers, and cursors on relational database.
(**Apply- L3**)
- CO3:** Create Unstructured Databases using MongoDB.(**Apply- L3**)
- CO 4:** Improve individual / teamwork skills, communication & report writing skills with ethical values.

Introduction: Language basics and example queries (**one or two weeks**)

1) Create a table STUDENT with appropriate data types and perform the following queries. Attributes are Roll number, student name, date of birth, branch and year of study.

- a) Insert 5 to 10 rows in a table?
- b) List all the students of all branches
- c) List student names whose name starts with 's'.
- d) List student names whose name contains 's' as third literal.
- e) List student names whose contains two 's' anywhere in the name
- f) List students whose branch is NULL.
- g) List students of CSE & ECE who born after 1980.
- h) List all students in reverse order of their names.
- i) Delete students of any branch whose name starts with 's'.
- j) Update the branch of CSE students to ECE.
- k) Display student name padded with '*' after the name of all the students.

2) Create the following tables based on the above Schema Diagram with appropriate data types

and constraints and perform the following queries.

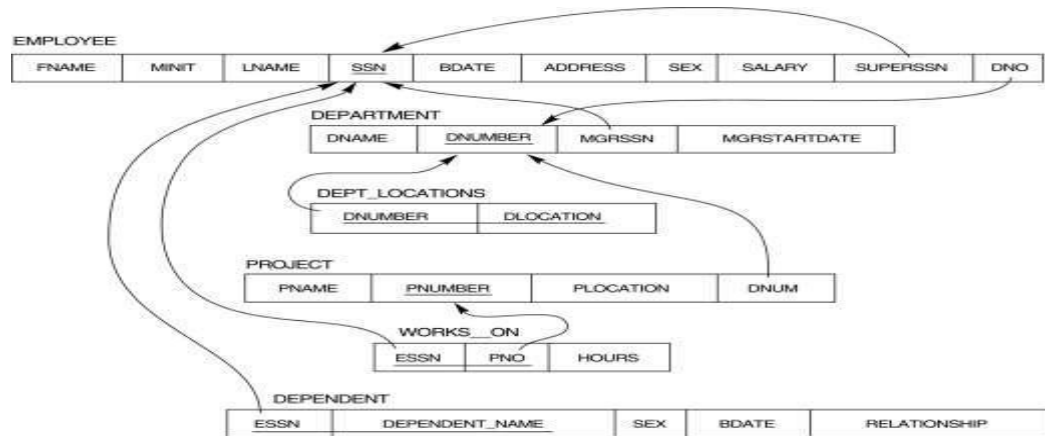
SAILORS (Sailid, Salname, Rating, Age)

RESERVES (Sailid, boatid, Day)

BOATS (Boatid, Boat-name, Color)

- a) Insert 5 to 10 rows in all tables?
- b) Find the name of sailors who reserved boat number 3.
- c) Find the name of sailors who reserved green boat.
- d) Find the colors of boats reserved by Ramesh.
- e) Find the names of sailors who have reserved at least one boat.
- f) Find the allsailid of sailors who have a rating of 10 or have reserved boated 104.

- g) Find the Sailid's of sailors with age over 20 who have not registered a red boat.
 - h) Find the names of sailors who have reserved a red or green boat.
 - i) Find sailors whose rating is better than some sailor called Salvador.
 - j) Find the names of sailors who are older than the oldest sailor with a rating of 10.
- 3) Schema Diagram for the rest of the SQL and PLSQL Programs.



Create the following tables based on the above Schema Diagram with appropriate data types and constraints.

EMPLOYEE (Fname, Mname, Lname, SSN, Bdate, Address, Gender, Salary, SuperSSN,Dno)

DEPARTMENT (Dnumber, Dname, MgrSSN, Mgrstartdate)

DEPENDENT (ESSN, Dependent_Name, Gender, Bdate, Relationship)

- a) Insert 5 to 10 rows into all the tables.
 - b) Display all employees' names along with their department names.
 - c) Display all employees' names along with their dependent details.
 - d) Display name and address of all employees who work for Research department.
 - e) List the names of all employees with two or more dependents.
 - f) List the names of employee who have no dependents.
 - g) List the names of employees who have at least one dependent.
 - h) List the names of the employees along with names of their supervisors using aliases.
 - i) Display name of the department and name of manager for all the departments.
 - j) Display the name of each employee who has a dependent with the same first name and gender as the employee.
- 4) Create the following tables based on the above Schema Diagram with appropriate data types and constraints in addition to the tables in Experiment 2.

DEPT_LOCATIONS (Dnumber, Dlocation)

PROJECT (Pname, Pnumber, Plocation,

Dnum) WORKS_ON (ESSN, Pno, Hours)

- a) Insert 5 to 10 rows into all the tables.
- b) Find the names of the employees who work on all the projects controlled by the department Research.

- c) List the project number, name and no. Of employees who work on that project for all the projects.
 - d) List the names of all the projects controlled by the departments department wise.
 - e) Retrieve the names of employees who work on all projects that John works on.
 - f) List the project numbers for projects that involve an employee either as worker or as a manager of the department that controls the project.
 - g) List the names of all employees in one department who work more than 10 hours on one specific project.
 - h) For each project, list the project name and total hours (by all employees) spent on that project.
 - i) Retrieve the names of all employees who work on every project.
 - j) Retrieve the names of all employees who do not work on any project.
- 5) Create a view that has project name, controlling department name, number of employees and total hours worked on the project for each project with more than one employee working on it.
- a) List the projects that are controlled by one department from this view.
 - b) List the managers of the controlling departments for all the projects.
 - c) Demonstrate one update operation on this view.
 - d) List the Location of the controlling departments for all the projects.
 - e) Retrieve the data from the view.
- 6) Create a view emp from employee such that it contains only emp_noemp_name and department.
- 7) Create a view dept from department with only dept_no and location.
- 8) Create a view that contains the details of employees who are managers only.
- 9) Write a procedure to check whether the given number is Armstrong or not.
- 10) Write a procedure which accept the account number of a customer and retrieve the balance.
- 11) Write a procedure which accepts the student number and displays the department in which he belongs to.
- 12) Create a cursor to modify the salary of all employees belonging to 'Research' department by 150%.
- 13) Consider the college database. Retrieve all students who have registered for a specific course and store their details into another table using Cursors.
- 14) Write an update trigger on Account table. The system should keep track of the records that are being updated.
- 15) Create NoSQL database for a sample application and perform CURD operations

Design Database for any one of the following Case Studies

Case Study1 : Hospital Management System

Aim: XYZ hospital is a multi-specialty hospital that includes a number of departments, rooms, doctors, nurses, compounders, and other staff working in the hospital. Patients having different kinds of ailments come to the hospital and get checkup done from the concerned doctors. If required they are admitted in the hospital and discharged after treatment. The aim of this case study is to design and develop a database for the hospital to maintain the records of various departments, rooms, and doctors in the hospital. It also maintains records of the regular patients, patients admitted in the hospital, the checkup of patients done by the doctors, the patients that have been operated, and patients discharged from the hospital.

Description: In hospital, there are many departments like Orthopedic, Pathology, Emergency, Dental, Gynecology, Anesthetics, I.C.U., Blood Bank, Operation Theater, Laboratory, M.R.I.,

Neurology, Cardiology, Cancer Department, Corpse, etc. There is an OPD where patients come and get a card (that is, entry card of the patient) for check up from the concerned doctor. After making entry in the card, they go to the concerned doctor's room and the doctor checks up their ailments. According to the ailments, the doctor either prescribes medicine or admits the patient in the concerned department. The patient may choose either private or general room according to his/her need. But before getting admission in the hospital, the patient has to fulfill certain formalities of the hospital like room charges, etc. After the treatment is completed, the doctor discharges the patient. Before discharging from the hospital, the patient again has to complete certain formalities of the hospital like balance charges, test charges, operation charges (if any), blood charges, doctors' charges, etc. Next we talk about the doctors of the hospital. There are two types of the doctors in the hospital, namely, regular doctors and call on doctors. Regular doctors are those doctors who come to the hospital daily. Calls on doctors are those doctors who are called by the hospital if the concerned doctor is not available.

Table Description:

Following are the tables along with constraints used in Hospital Management database

1. DEPARTMENT: This table consists of details about the various departments in the hospital. The information stored in this table includes department name, department location, and facilities available in that department.

Constraint: Department name will be unique for each department.

2. ALL_DOCTORS: This table stores information about all the doctors working for the hospital and the departments they are associated with. Each doctor is given an identity number starting with DR or DC prefixes only.

Constraint: Identity number is unique for each doctor and the corresponding department should exist in DEPARTMENT table.

3. DOC_REG: This table stores details of regular doctors working in the hospital. Doctors are referred to by their doctor number. This table also stores personal details of doctors like name, qualification, address, phone number, salary, date of joining, etc. Constraint: Doctor's number entered should contain DR only as a prefix and must exist in ALL_DOCTORS table.

4. DOC_ON_CALL: This table stores details of doctors called by hospital when additional doctors are required. Doctors are referred to by their doctor number. Other personal details like name, qualification, fees per call, payment due, address, phone number, etc., are also stored.

Constraint: Doctor's number entered should contain DC only as a prefix and must exist in ALL_DOCTORS table.

5. PAT_ENTRY: The record in this table is created when any patient arrives in the hospital for a checkup. When patient arrives, a patient number is generated which acts as a primary key. Other details like name, age, sex, address, city, phone number, entry date, name of the doctor referred to, diagnosis, and department name are also stored. After storing the necessary details patient is sent to the doctor for checkup.

Constraint: Patient number should begin with prefix PT. Sex should be M or F only. Doctor's name and department referred must exist.

6. PAT_CHKUP: This table stores the details about the patients who get treatment from the doctor referred to. Details like patient number from patient entry table, doctor number, date of checkup, diagnosis, and treatment are stored. One more field status is used to indicate whether patient is admitted, referred for operation or is a regular patient to the hospital. If patient is admitted, further details are stored in PAT_ADMIT table. If patient is referred for operation, the further details are stored in PAT_OPR table and if patient is a regular patient to the hospital, the further details are stored in PAT_REG table.

Constraint: Patient number should exist in PAT_ENTRY table and it should be unique.

7. PAT_ADMIT: When patient is admitted, his/her related details are stored in this table. Information stored includes patient number, advance payment, mode of payment, room number, department, date of admission, initial condition, diagnosis, treatment, number of the doctor under whom treatment is done, attendant name, etc.

Constraint: Patient number should exist in PAT_ENTRY table. Department, doctor number, room number must be valid.

8. PAT_DIS: An entry is made in this table whenever a patient gets discharged from the hospital. Each entry includes details like patient number, treatment given, treatment advice, payment made, mode of payment, date of discharge, etc.

Constraint: Patient number should exist in PAT_ENTRY table.

9. PAT_REG: Details of regular patients are stored in this table. Information stored includes date of visit, diagnosis, treatment, medicine recommended, status of treatment, etc.

Constraint: Patient number should exist in patient entry table. There can be multiple entries of one patient as patient might be visiting hospital repeatedly for checkup and there will be entry for patient's each visit.

10. PAT_OPR: If patient is operated in the hospital, his/her details are stored in this table. Information stored includes patient number, date of admission, date of operation, number of the doctor who conducted the operation, number of the operation theater in which operation was carried out, type of operation, patient's condition before and after operation, treatment advice, etc.

Constraint: Patient number should exist in PAT_ENTRY table. Department, doctor number should exist or should be valid.

11. ROOM_DETAILS: It contains details of all rooms in the hospital. The details stored in this table include room number, room type (general or private), status (whether occupied or not), if occupied, then patient number, patient name, charges per day, etc.

Constraint: Room number should be unique. Room type can only be G or P and status can only be Y or N.

Case Study2 :Railway Reservation

Aim: The railway reservation system facilitates the passengers to enquire about the trains available on the basis of source and destination, booking and cancellation of tickets, enquire about the status of the booked ticket, etc. The aim of case study is to design and develop a database maintaining the records of different trains, train status, and passengers. The record of train includes its number, name, source, destination, and days on which it is available, whereas

record of train status includes dates for which tickets can be booked, total number of seats available, and number of seats already booked. The database has been developed and tested on the Oracle.

Description:

Passengers can book their tickets for the train in which seats are available. For this, passenger must provide the desired train number and the date for which ticket is to be booked. Before booking a ticket for a passenger, the validity of train number and booking date is checked. Once the train number and booking date are validated, it is checked whether the seat is available. If yes, the ticket is booked with confirm status and corresponding ticket ID is generated which is stored along with other details of the passenger. After all the available tickets are booked, certain numbers of tickets are booked with waiting status. If waiting lot is also finished, then tickets are not booked and a message of non-availability of seats is displayed. The ticket once booked can be cancelled at any time. For this, the passenger must provide the ticket ID (the unique key). The ticket ID is searched, and the corresponding record is deleted. With this, the first ticket with waiting status also gets confirmed.

List of Assumption

Since the reservation system is very large in reality, it is not feasible to develop the case study to that extent and prepare documentation at that level. Therefore, a small sample case study has been created to demonstrate the working of the reservation system. To implement this sample case study, some assumptions have been made, which are as follows:

1. The number of trains has been restricted to 5.
2. The booking is open only for next seven days from the current date.
3. Only two categories of tickets can be booked, namely, AC and General.
4. The total number of tickets that can be booked in each category (AC and General) is 10.
5. The total number of tickets that can be given the status of waiting is 2.
6. The in-between stoppage stations and their bookings are not considered.

Description of Tables and Procedures

Tables and procedures that will be created are as follows:

1. TrainList: This table consists of details about all the available trains. The information stored in this table includes train number, train name, source, destination, fair for AC ticket, fair for general ticket, and weekdays on which train is available.

Constraint: The train number is unique.

2. Train_Status: This table consists of details about the dates on which ticket can be booked for a train and the status of the availability of tickets. The information stored in this table includes train number, train date, total number of AC seats, total number of general seats, number of AC seats booked, and number of general seats booked.

Constraint: Train number should exist in TrainList table.

3. Passenger: This table consists of details about the booked tickets. The information stored in this table includes ticket ID, train number, date for which ticket is booked, name, age, sex and

address of the passenger, status of reservation (either confirmed or waiting), and category for which ticket is booked.

Constraint: Ticket ID is unique and the train number should exist in TrainList table.

4.Booking: In this procedure, the train number, train date, and category is read from the passenger. On the basis of the values provided by the passenger, corresponding record is retrieved from the Train_Status table. If the desired category is AC, then total number of AC seats and number of booked AC seats are compared in order to find whether ticket can be booked or not. Similarly, it can be checked for the general category. If ticket can be booked, then passenger details are read and stored in the Passenger table.

5.Cancel: In this procedure, ticket ID is read from the passenger and corresponding record is searched in the Passenger table. If the record exists, it is deleted from the table. After deleting the record (if it is confirmed), first record with waiting status for the same train and same category are searched from the Passenger table and its status is changed to confirm.

Case Study3 :Painting Hire Business

System Description:

A local businesswoman has decided to start her own Internet business, called MasterpiecesLtd, hiring paintings to private individuals and commercial companies. Because of your reputation as a database designer, she has called upon your services to design and implement a database to support her new business. At the initial planning meeting, to discuss the design, the following user requirements were requested. The system must be able to manage the details of customers, paintings and those paintings currently on hire to customers. Customers are categorized as B (bronze), S (silver), G (gold) or P (platinum). These categories entitle a customer to a discount of 0%, 5%, 10% or 15% respectively.

Customers often request paintings by a particular artist or theme (e.g. animal, landscape, seascape, naval, still-life, etc). Over time a customer may hire the same painting more than once.

Each painting is allocated a customer monthly rental price defined by the owner. The owner of the painting is then paid 10% of that customer rental price. Any paintings that are not hired within six months are returned to the owner. However, after three months, an owner may resubmit a returned painting. Each painting can only have one artist associated with it. Several reports are required from the system. Three main ones are:

1. For each customer, a report showing an overview of all the paintings they have hired or are currently hiring
2. For each artist, a report of all paintings submitted for hire
3. For each artist, a returns report for those paintings not hired over the past six months
remember to identify key attributes and any foreign key attributes.

B.Tech. (III Sem.)

**20CS57 - OBJECT ORIENTED PROGRAMMING
LAB**

| L | T | P | Cr. |
|---|---|---|-----|
| 0 | 0 | 3 | 1.5 |

Pre-requisite : Programming for Problem Solving using C and Data structures

Course Educational Objective:The objective of the course is to apply the constructs of Java programming language along with built-in facilities to create different applications such as console & graphical user interfaces. They will be applying knowledge of object-oriented programming, collection framework to perform all operations on data.

Course Outcomes (CO): *At the end of this course, the student will be able to:*

- CO1:** Solve basic mathematical problems using fundamentals of Java and its object-oriented principles. **(Apply – L3)**
- CO2:** Implement multithreading and exception handling mechanisms. **(Apply – L3)**
- CO3:** Develop GUI applications and basic data structures using collection framework. **(Apply – L3)**
- CO 4:** Improve individual / teamwork skills, communication & report writing skills with ethical values.

Introduction: Language basics and example problems **(one or two Modules)**

1. Write a C++ program to display default value of all primitive data type of Java.
2. Write a C++ program that display the roots of a quadratic equation $ax^2+bx=0$. Calculate the discriminate D and basing on value of D, describe the nature of root.
3. Write a C++ Program to find the factorial of list of numbers reading input as command line argument.
4. Write a C++ Program to display all prime numbers between two limits.
5. Five bikers compete in a race such that they drive at a constant speed, which may or may not be the same as the other. To qualify the race, the speed of a racer must be more than the average speed of all five racers. Take as input the speed of each racer and print back the speed of qualifying racers.
6. Write a C++ program to implement Rhombus pattern reading the limit from the user.
7. Write a C++ program to calculate gross salary & net salary taking the following data.
Input: empno, empname, basic salary
Process: DA=50% of basic, HRA=25% of basic, CCA=Rs240/-, PF=10% of basic, PT=Rs100/-.
8. Write a C++ program to search for an element in a given list of elements using binary search mechanism.

Module 1:

- a. Develop a java program to create class, which contains data & methods, create an object to access those members.
- b. Develop a java program, which implements all types of java variables (local, class level: static, instance).
- c. Develop a java program to calculate the sum of diagonal elements of given n x n matrix.

Module 2:

- a. Develop a java program, which contains both static and non-static methods.
- b. Develop a java program to find area of geometrical figures using method.
- c. Develop a java program to initialize instance variables by using constructors.
- d. Develop a java program, which implements constructor overloading by passing different number of parameters of different types.

Module 3:

- a. Develop a java program to count the words, characters in the given line of text.
- b. Develop a java program for sorting a given list of names in ascending order.
- c. Develop a java program that reads a line of integers separated by commas and then displays each integer, find the sum of the integers (using StringTokenizer).
- d. Develop a java program to implement multi-level inheritance.

Module 4:

- a. Develop a java program to create and access user-defined package.
- b. Develop a java program to identify the accessibility of a variable by means of different access specifier within and outside the package.
- c. Develop a java program to implement the concept of method overloading.
- d. Develop a java program to implement the concept of method overriding.

Module 5:

- a. Develop a java program for abstract class to find areas of different shapes.
- b. Develop a java program to achieve multiple inheritance using interfaces.
- c. Develop a java program to create an interface named Vehicle which contains two abstract methods (Specifications (), Display ()). Provide two classes named Two-wheeler, Four-wheeler that is implemented by that interface.

Module 6:

- a. Develop a java program that implements a multi-threaded program, which has three threads. First thread generates a random integer for every 1 second, if the generated integer is even the second thread computes the square of the number and print it. If the generated integer is odd the third thread will print the value of cube of the number.
- b. Develop a java program to identify the use of synchronized blocks, synchronized methods and static synchronized methods in threads concept.
- c. Develop a java program to illustrate the concept of inter thread communication.

Module 7:

- a. Develop a java program that creates a user interface to perform integer divisions with possible validations (Divide by Zero, NumberFormatException).
- b. Develop a java program to implement mouse events like mouse pressed, mouse released, and mouse moved by means of adapter classes.

Module 8:

- a. Develop a java program that works as a simple calculator. Use a Grid Layout to arrange Buttons for digits and for the + - * % operations. Add a text field to display the result. Handle any possible exceptions like divide by zero.
- b. Develop a java program to simulate a traffic light, user can select any one of the three buttons with: red, yellow, and green color. On selecting a button, an appropriate message with “**Stop**” or “**Ready**” or “**Go**” should appear with the selected button color.

Module 9:

- a. Develop a java program to print the collection data by using the following ways
i) for loop ii) for-each loop iii) Iterator iv) ListIterator
- b. Develop a java program to perform all the operations in Collection interface.

Module 10:

- a. Develop a java program to implement and perform all the operations in List, Set Interface.
- b. Develop a java program to implement and perform all the operations in Map interface.

B.Tech. (III Sem.)

20IT53 – R PROGRAMMING LAB

| L | T | P | Cr. |
|---|---|---|-----|
| 0 | 0 | 3 | 1.5 |

Pre-requisite : Nil

Course Educational Objective: In this course student will learn about the fundamentals of R programming, standard R libraries, solid understanding of R functions, write programs using the R and gain skills in R programming Language, get acquaintances with Arrays, Files, Strings, Packages, and distributions using R.

Course Outcomes: At the end of this course,

- CO1:** Implement basic concepts of R programming, and its different module that includes conditional, looping, lists, Strings, Functions, Frames, Arrays, and File programming. (**Understand - L2**)
- CO2:** Implement the concepts of R Script to extract the data from data frames and file operations. (**Apply – L3**)
- CO3:** Implement the various statistical techniques using R. (**Understand - L2**)
- CO4:** Extend the functionality of R by using add-on packages. (**Understand - L2**)
- CO5:** Use R Graphics and Tables to visualize results of various statistical operations on data. (**Apply – L3**)

Week 1:

- a) Installing R and RStudio
- b) Basic functionality of R, variable, data types in R

Week 2:

- a) Implement R script to show the usage of various operators available in R language.
- b) Implement R script to read person's age from keyboard and display whether he is eligible for voting or not.
- c) Implement R script to find biggest number between two numbers.
- d) Implement R script to check the given year is leap year or not.

Week 3:

- a) Implement R Script to create a list.
- b) Implement R Script to access elements in the list.
- c) Implement R Script to merge two or more lists. Implement R Script to perform matrix operation

Week 4:

Implement R script to perform following operations:

- a) various operations on vectors
- b) Finding the sum and average of given numbers using arrays.
- c) To display elements of list in reverse order.
- d) Finding the minimum and maximum elements in the array.

Week 5:

- a) Implement R Script to perform various operations on matrices
- b) Implement R Script to extract the data from dataframes.
- c) Write R script to display file contents.
- d) Write R script to copy file contents from one file to another

Week 6:

- a) Write an R script to find basic descriptive statistics using summary, str, quartile function onmtcars & cars datasets.
- b) Write an R script to find subset of dataset by using subset (), aggregate () functions on iris dataset

Week 7:

- a) Reading different types of data sets (.txt, .csv) from Web or disk and writing in file in specific disk location.
- b) Reading Excel data sheet in R.
- c) Reading XML dataset in R

Week 8:

- a) Implement R Script to create a Pie chart, Bar Chart, scatter plot and Histogram (Introduction to ggplot2 graphics)
- b) Implement R Script to perform mean, median, mode, range, summary, variance, standard deviation operations.

Week 9:

- a) Implement R Script to perform Normal, Binomial distributions.
- b) Implement R Script to perform correlation, Linear and multiple regression.

Week 10:

Introduction to Non-Tabular Data Types: Time series, spatial data, Network data.
Data Transformations: Converting Numeric Variables into Factors, Date Operations, String Parsing, Geocoding.

Week 11:

Introduction Dirty data problems: Missing values, data manipulation, duplicates, forms of data dates, outliers, spelling.

Week 12:

Data sources: SQLite examples for relational databases, Loading SPSS and SAS files, Reading from Google Spreadsheets, API and web scraping examples

Reference books:

1. R Cookbook Paperback – 2011 by Teetor Paul O Reilly Publications
2. Beginning R: The Statistical Programming Language by Dr. Mark Gardener, Wiley Publications.
3. R Programming For Dummies by JorisMeysAndrie de Vries, Wiley Publications
4. Hands-On Programming with R by Grolemund, O Reilly Publications
5. Statistical Programming in R by KG Srinivas G.M. Siddesh, ChetanShetty&Sowmya B.J. – 2017 edition
6. R Fundamentals and Programming Techniques, ThomasLumely.
7. R for Everyone Advanced Analytics and Graphics, Jared P. Lander- Addison Wesley Series.
8. The Art of R Programming, Norman Matloff, Cengage Learning.
9. Maria Dolores Ugarte, Ana F.Militino, AlanT.Arnholt—Probability and Statistics with R, 2nd Edition, CRC Press,2016.
10. R-programming for Data science, Roger D.Peng.
11. An Introduction to statistical learning-with applications in R, Trevor Hastie and Rob Tibshirani.

Web Links:

1. URL: <https://cran.r-project.org/doc/manuals/r-release/R-intro.pdf> (Online Resources)
2. <http://nptel.ac.in/courses/106104135/48>
3. <http://nptel.ac.in/courses/110106064/>

Software requirements:

1. The R statistical software program. Available from: <https://www.r-project.org/>
2. RStudio an Integrated Development Environment (IDE) for R. Available from:<https://www.rstudio.com/>

B.Tech. (III Sem.)

**20CSS1–WEB APPLICATION DEVELOPMENT
USING FULL STACK – MODULE - I
(FRONT END DEVELOPMENT)**

| L | T | P | Cr. |
|---|---|---|-----|
| 1 | 0 | 2 | 2 |

Pre-requisite : Knowledge of basic Computer hardware & software.

Course Educational Objective:The objective of the course is to understand the design of HTML web pages, Styling of HTML pages using CSS, web forms validation using JavaScript and developing responsive web page using JQuery.

Course Outcomes (CO): *At the end of this course, the student will be able to:*

- CO1:** Understand the basic design and styling of web pages(**Understand- L2**)
- CO2:** Understand the DOM of web design, markup language and client-side scripting. (**Understand- L2**)
- CO3:** Understand the responsive web design using DHTML. (**Understand- L2**)
- CO 4:** Improve individual / teamwork skills, communication & report writing skills with ethical values.

Modules

1. Design of web pages
2. Creating Web applications using HTML
3. Styling the web pages using CSS
4. Validating the web forms using javaScript
5. Handling and validating semi-structured data
 - a. XML
 - b. JSON
6. jQuery
 - a. validation
 - b. selectors
 - c. fades
 - d. slide
 - e. hide
 - f. animation

REFERENCE BOOKS:

1. HTML & CSS: The Complete Reference, 5thEditionby Thomas Powell, McGrawHill, 2017.
2. Beginning HTML, XHTML, CSS, and JavaScriptby Jon Duckett, Wiley India, 2010.
3. jQuery Cookbook by Cody Lindley, O'Reilly Media, 2009
4. HTML, XHTML, and CSS Bible, 5th Edition by Steven M. Schafer, Wiley India, 2011.
5. Web Development with jQuery by Richard York, Wiley India, 2015.

B.Tech. (IV Sem.)

**20CS06 - DESIGN AND ANALYSIS OF
ALGORITHMS**

| L | T | P | Cr. |
|---|---|---|-----|
| 3 | 0 | 0 | 3 |

Pre-requisite : Discrete Mathematical Structures and Data Structures.

Course Educational Objective: The Objective of the course is to learn various algorithm design techniques and analyze the computing resources of the algorithms ,and motivate the students to design new algorithms for various problems

Course Outcomes: At the end of this course, the student will be able to

- CO1:** Identify the characteristics of an algorithm and analyze its time and space complexity. **(Understand- L2)**
- CO2:** Apply the divide-and-conquer method for solving problems like searching and sorting. **(Apply- L3)**
- CO3:** Design Greedy algorithms for the optimization problems like knapsack problem, minimum cost spanning tree, single source shortest path problem. **(Apply - L3)**
- CO4:** Apply dynamic programming paradigm to solve optimization problems like travelling salesperson problem,0/1 knapsack problem, Optimal binary search tree. **(Apply - L3)**
- CO5:** Analyze the backtracking and branch-and-bound search methods on optimization problems like N-queens, sum of subsets,0/1 knapsack, Hamiltonian circuit and so on. **(Apply - L3)**

UNIT – I

Introduction: Algorithm definition, Specifications, Performance Analysis- Time Complexity, Space Complexity. Asymptotic Notations-Big-Oh, Omega, Theta.

Divide and Conquer: General Method, Binary Search, Finding Maximum and Minimum, Merge Sort, Quick sort, closest pair of points.

UNIT – II

The Greedy Method – General Method, Knapsack Problem, Job sequencing with deadlines, Minimum-cost spanning trees, Optimal storage on tapes, Single source shortest paths, Huffman coding.

UNIT – III

Dynamic Programming - General method, Multistage graph, All pairs shortest path, Single Source Shortest path, Optimal Binary search trees, 0/1 Knapsack, Reliability design, the travelling salesman problem.

UNIT - IV

Back tracking - The General Method, The 8-Queens Problem, Sum of subsets, Graph Coloring, Hamiltonian cycles.

UNIT-V

Branch and Bound – General method, Job sequencing with deadlines –LC Branch and Bound, FIFO Branch and Bound and LIFO Branch and Bound, 0/1 Knapsack problem - LC Branch and Bound solution, FIFO Branch and Bound solution, Travelling salesperson Problem – LC Branch and Bound solution

TEXTBOOK(S):

1. Ellis Horowitz, Sartaj Sahni, S Rajasekaran , “Fundamentals of Computer Algorithms”, University press, 2nd edition, 2012.

REFERENCE BOOKS:

1. Mark Allen Weiss, “Data Structures and Algorithm Analysis in C++”, Pearson, 3rd edition , 2007.
2. Aho, Hopcroft & Ullman, “The Design and Analysis of Computer Algorithms”, Addison Wesley publications, 2008.
3. Thomas H. Corman et al, “Introduction to Algorithms”, PHI, 3rd edition, 2008
4. Anany Levitin, “Introduction to the Design and Analysis of Algorithms”, PEA,
5. P. H. Dave, H. B. Dave, “Design and Analysis of Algorithms”, Pearson Education”, 2008.

B.Tech. (IV Sem.)

20CS10 - DATA WAREHOUSING & DATA MINING

| L | T | P | Cr. |
|---|---|---|-----|
| 3 | 0 | 0 | 3 |

Pre-requisite : DBMS and Probability and Statistics

Course Educational Objective: The Objective of the course is to introduce the concepts of data warehouse and data mining, which gives a complete description about the principles, used, architectures, applications, design and implementation of data mining and data ware housing concepts.

Course Outcomes: At the end of this course, the student will be able to

- CO1:** Summarize the architecture of data warehouse.(**Understand- L2**)
- CO2:** Apply different preprocessing methods, Similarity, Dissimilarity measures for any given raw data.(**Apply – L3**)
- CO3:** Construct a decision tree and resolve the problem of model overfitting.(**Analyze– L4**)
- CO4:** Compare Apriori and FP-growth association rule mining algorithms for frequent itemset generation.(**Apply - L3**)
- CO5:** Apply suitable clustering algorithm for the given data set.(**Apply - L3**)

UNIT-I

Data Warehouse and OLAP Technology: An Overview: Data Warehouse, A Multidimensional DataModel, Data Warehouse Architecture, Data Warehouse Implementation, From Data Warehousing to DataMining. (Han &Kamber)

UNIT-2

Data Mining: Introduction, Data Mining, Motivating challenges, The origins of Data Mining, Data MiningTasks, Types of Data, Data Quality.
Data Preprocessing: Aggregation, Sampling, Dimensionality Reduction, Feature Subset Selection, Featurecreation, Discretization and Binarization, Variable Transformation, Measures of Similarity andDissimilarity. (Tan &Vipin)

UNIT-3

Classification: Basic Concepts, General Approach to solving a classification problem, Decision TreeInduction: Working of Decision Tree, building a decision tree, methods for expressing an attribute testconditions, measures for selecting the best split, Algorithm for decision tree induction.

Model Overfitting: Due to presence of noise, due to lack of representation samples, evaluating theperformance of classifier: holdout method, random sub sampling, cross-validation, bootstrap. BayesTheorem, Naïve Bayes Classifier (Tan &Vipin)

UNIT-4

Association Analysis: Basic Concepts and Algorithms: Problem Definition, Frequent Item Set Generation,Apriori Principle, Apriori Algorithm, Rule Generation, Compact Representation of

Frequent Itemsets, FPGrowthAlgorithm. (Tan & Vipin)

UNIT-5

Cluster Analysis: Basic Concepts and Algorithms: Overview, What Is Cluster Analysis? Different Types of Clustering, Different Types of Clusters; K-means: The Basic K-means Algorithm, K-means Additional Issues, Bisecting K-means, Strengths and Weaknesses; Agglomerative Hierarchical Clustering: Basic Agglomerative Hierarchical Clustering Algorithm DBSCAN: Traditional Density Center-Based Approach, DBSCAN Algorithm, Strengths and Weaknesses. (Tan & Vipin)

TEXTBOOKS:

1. Introduction to Data Mining : Pang-Ning Tan & Michael Steinbach, Vipin Kumar, Fifth Impression, Pearson, 2015.
2. Data Mining concepts and Techniques, 3rd Edition, Jiawei Han, Michel Kamber, Elsevier, 2011

REFERENCE BOOKS:

1. Data Mining Techniques and Applications: An Introduction, Hongbo Du, Cengage Learning, 2010.
2. Data Mining : Introductory and Advanced topics : Dunham, First Edition, Pearson, 2020
3. Data Warehousing Data Mining & OLAP, Alex Berson, Stephen Smith, TMH, 2008.
4. Data Mining Techniques, Arun K Pujari, Universities Press, 2001

B.Tech. (IV Sem.)

20CS11 - OPERATING SYSTEMS

| L | T | P | Cr. |
|---|---|---|-----|
| 3 | 0 | 0 | 3 |

Pre-requisite : Knowledge of Computer fundamentals & Data structures & Algorithms

Course Educational Objective: The objective of the course is to provide basic knowledge of computer operating system structure and functioning, understand how Operating Systems evolved with advent of computer architecture, and comprehend the different CPU scheduling algorithms, page replacement algorithms, disk scheduling and identify best one.

Course Outcomes: At the end of this course, the student will be able to

- CO1:** Demonstrate the underlying principles and techniques of operating system(**Understand-L2**)
- CO2:** Interpret scheduling and communication methods of processes handled by operating systems(**Understand-L2**).
- CO3:** Distinguish the process synchronization methods and deadlock handling approaches employed in operating systems(**Understand-L2**).
- CO4:** Classify memory management techniques and virtual memory mechanisms(**Understand-L2**).
- CO5:** Interpret the strategies of disk scheduling algorithms and file system architecture (**Understand-L2**).

Unit-1: Introduction to Operating System

Operating System Structures: Operating-System Services, User Operating-System Interface, System Calls, Types of System Calls, System Programs, Operating-System Design and Implementation, Operating-System Structure, Virtual Machines, Operating-System Generation, System Boot.

Unit-2: Process Management

Processes: Process concept, Inter-process Communication, Communication in Client-Server Systems. **Threads:** Overview, Multithreading Models

Process Scheduling: Scheduling Criteria, Scheduling Algorithms (FCFS, SJF, PRIORITY, ROUNDROBIN)

Unit-3: Synchronization and Deadlocks

Synchronization: The Critical-Section Problem, Peterson's Solution, Synchronization Hardware, Semaphores, Classic Problems of Synchronization, Monitors.

Deadlocks: System Model, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention. Deadlock Avoidance, Deadlock Detection, Recovery from deadlock.

Unit-4: Memory Management

Memory Management Strategies: Swapping, Contiguous Memory Allocation, Paging, Structure of the Page Table, Segmentation.

Virtual Memory Management: Demand Paging, Page Replacement, Allocation of Frames, Thrashing.

Unit-5: File System Management

Mass-Storage Structure: Overview of Mass-Storage Structure, Disk Structure, Disk Attachment, Disk Scheduling, Disk Management.

Implementing File System: File-System Structure, File-System Implementation, Directory Implementation, Allocation Methods, Free-Space Management, Efficiency and Performance, Recovery.

TEXTBOOKS:

1. Silberschatz & Galvin, "Operating System Concepts", Wiley, 7th edition, 2007.

REFERENCE BOOKS:

1. William Stallings, "Operating Systems", PHI, 5th Edition, 2004.
2. B.A. Forouzan & R.F. Giberg, —Unix and shell Programming, Thomson, First Edition, New Delhi, 2003.
3. <http://codex.cs.yale.edu/avi/os-book/OS9/slide-dir/index.html>
4. https://swayam.gov.in/nd1_noc19_cs50/preview

B.Tech. (IV Sem.)

20IT01 - SOFTWARE ENGINEERING

| L | T | P | Cr. |
|---|---|---|-----|
| 3 | 0 | 0 | 3 |

Pre-requisite : Object Oriented Programming

Course Educational Objective: The objective of the course is to provide understanding of different s/w process models and how to choose one among them by gathering the requirements from a client and specifying them. Using those requirements in the design of the software architecture based on the choices with the help of modules and interfaces. To enable s/w development, by using different testing techniques like unit, integration and functional testing, the quality assurance can be achieved.

Course Outcomes: At the end of this course, the student will be able to

- CO 1 Understand the fundamentals of software engineering concepts and software process models. **(Understand-L2)**
- CO 2 Apply the requirement elicitation techniques for preparing SRS and design engineering. **(Apply-L3)**
- CO 3 Understanding the basic building blocks of UML, Class and object diagrams. **(Understand-L2)**
- CO 4 Apply the behavioral models for real world applications. **(Apply-L3)**
- CO 5 Demonstrate different software testing approaches for testing the real time applications. **(Understand-L2)**

UNIT – I:

Software and software Engineering: The evolving role of Software, Characteristics of Software, Importance of software Engineering, Changing nature of software, Legacy Software, Software Myths.

Software Process and Process Models: Layered technology, Process frame work, The process and Product, software process models, the water fall model, incremental model, the spiral and V Model, Component based s/w development, Unified process model,

UNIT – II:

Requirements Analysis and Software design: Requirements gathering and analysis, software requirements specifications (SRS).

Design Engineering: overview of design process, Design Concepts, Architectural Concepts

UNIT – III:

Design Using UML: Building Blocks of UML, Defining things, relationships and diagrams, Common Mechanism in UML, Class and Object Diagrams

UNIT – IV:

Behavioral Modeling: Interactions, Interaction diagrams, use cases, Use case Diagrams, Activity Diagrams, Events and signals, state machines, processes and Threads, time and space, state chart diagrams

Architectural Modeling: Component, Deployment, Component diagrams and Deployment diagrams.

UNIT – V:

Testing Techniques: Software testing fundamentals, Unit testing, Integration testing, Black box testing, white box testing, Debugging, System testing.

TEXTBOOKS:

1. Roger S. Pressman, “Software engineering- A practitioner ‘s Approach”, TMH International Edition, 6th edition, 2005.
2. Grady Booch, James Rumbaugh, Ivar Jacobson, “The Unified Modelling Language User Guide”, PEARSON ,4th Impression, 2012.

REFERENCE BOOKS:

1. Software Engineering - Concepts and practices: Ugrasen Suman, Cengage learning
2. Object-oriented analysis and design using UML”, Mahesh P. Matha, PHI
3. Fundamentals of Software Engineering, Rajib Mall, Third Edition ,PHI
4. https://onlinecourses.nptel.ac.in/noc20_cs68 [1,2,3,4,5]

B.Tech. (IV Sem.)

**20HS01 – UNIVERSAL HUMAN VALUES 2:
UNDERSTANDING HARMONY**

| L | T | P | Cr. |
|---|---|---|-----|
| 3 | 0 | 0 | 3 |

Pre-requisite : Nil

Course Educational Objective: The objective of the course is to become more aware of themselves, and their surroundings (family, society, nature); they would become more responsible in life, and in handling problems with sustainable solutions, while keeping human relationships and human nature in mind.

Course Outcomes: At the end of this course,

- CO1:** Apply the value inputs in life and profession (**Apply – L3**)
- CO2:** Distinguish between values and skills, happiness and accumulation of physical facilities, the self, and the Body (**Understand – L2**)
- CO3:** Understand the role of a human being in ensuring harmony in society (**Understand – L2**).
- CO4:** Understand the role of a human being in ensuring harmony in the nature and existence. (**Understand – L2**)
- CO5:** Distinguish between ethical and unethical practices (**Apply – L3**)

UNIT – I: Need, Basic Guidelines, Content and Process for Value Education

‘Natural Acceptance’ and Experiential Validation- as the process for self-exploration; Continuous Happiness and Prosperity- A look at basic Human Aspirations; Right understanding, Relationship and Physical Facility, Understanding Happiness and Prosperity

UNIT – II: Understanding Harmony in the Human Being - Harmony in Myself!

Understanding human being as a co-existence of the sentient ‘I’ and the material ‘Body’; Understanding the needs of Self (‘I’) and ‘Body’ - happiness and physical facility; Understanding the Body as an instrument of ‘I’ (I am being the doer, seer and enjoyer);

Understanding the characteristics and activities of ‘I’ and harmony in ‘I’; Understanding the harmony of I with the Body: Sanyam and Health; correct appraisal of Physical needs, meaning of Prosperity in detail

UNIT – III: Understanding Harmony in the Family and Society- Harmony in Human-Human Relationship

Understanding values in human-human relationship; meaning of Justice (nine universal values in relationships) and program for its fulfilment to ensure mutual happiness; Trust and Respect as the foundational values of relationship.

Understanding the harmony in the society: Resolution, Prosperity, fearlessness and co-existence as comprehensive Human Goals; Visualizing a universal harmonious order in society- Undivided

Society, Universal Order- from family to world family, Gratitude as a universal value in relationships.

UNIT – IV: Understanding Harmony in the Nature and Existence - Whole existence as Coexistence

Understanding the harmony in the Nature; Interconnectedness and mutual fulfilment among the four orders of nature- recyclability and self-regulation in nature; Understanding Existence as Co-existence of mutually interacting units in all-pervasive space; Holistic perception of harmony at all levels of existence.

UNIT – V: Implications of the above Holistic Understanding of Harmony on Professional Ethics

Natural acceptance of human values; Definitiveness of Ethical Human Conduct; Basis for Humanistic Education, Humanistic Constitution and Humanistic Universal Order; Competence in professional ethics, Strategy for transition from the present state to Universal Human Order

TEXTBOOK(S):

1. Human Values and Professional Ethics by R R Gaur, R Sangal, G P Bagaria, Excel Books, New Delhi, 2010

REFERENCE BOOKS:

1. Jeevan Vidya: EkParichaya, A Nagaraj, Jeevan Vidya Prakashan, Amarkantak, 1999.
2. Human Values, A.N. Tripathi, New Age Intl. Publishers, New Delhi, 2004.
3. The Story of My Experiments with Truth - by Mohandas Karamchand Gandhi

B.Tech. (IV Sem.)

20MC02 – ENVIRONMENTAL SCIENCE

| L | T | P | Cr. |
|---|---|---|-----|
| 2 | 0 | 0 | 0 |

Pre-requisite : Nil

Course Educational Objective: The objective of this course is to understand Environmental issues like over population, human health etc., related to local, regional, and global levels. The necessity of resources, their exploitation and sustainable management. The interactions of human and ecosystems and their role in the food web in the natural world. The global biodiversity, threats to biodiversity and its conservation. Environmental problems like pollution, disasters and possible solutions. The importance of environmental decision making in organizations through audits.

Course Outcomes: At the end of this course, the student will be able to

- CO1:** Identify environmental problems arising due to engineering and technological activities that help to be the part of sustainable solutions. (**Remember– L1**)
- CO2:** Evaluate local, regional, and global environmental issues related to resources and their sustainable management (**Understand - L2**)
- CO3:** Realize the importance of ecosystem and biodiversity for maintaining ecological balance. (**Understand - L2**)
- CO4:** Acknowledge and prevent the problems related to pollution of air, water, and soil. (**Apply– L3**)
- CO5:** Identify the significance of implementing environmental laws and abatement devices for environmental management. (**Understand - L2**)

UNIT – I :Nature and scope of Environmental Problems

- Introduction to Environment Science.
- Population explosion, variations among nations
- Resettlement and Rehabilitation - Issues and possible solutions
- Environmental hazards – causes and solutions. Biological hazards – AIDS, Malaria, Chemical hazards- BPA, PCB, Phthalates, Mercury, Nuclear hazards- Risk and evaluation of hazards.
- Role of Information Technology in environmental management and human health

UNIT – II :Natural Resources and Conservation

Introduction and classification of Natural Resources

- Forest resources: Use and over-exploitation, deforestation, Timber extraction, mining, dams and their effects on forests and tribal people
- Water resources: Use and over-utilization of surface and ground water, conflicts over water, interlinking of rivers, dams-benefits and problems, Rainwater harvesting
- Mineral resources: Use and exploitation, environmental effects of extracting and using mineral resources
- Food resources: World food problems, changes caused by agriculture and overgrazing, effects of modern agriculture, fertilizer-pesticide problems, water logging, soil salinity
- Energy resources: Growing energy needs renewable, non-renewable and alternate energy resources

UNIT – III :Ecology and Biodiversity

- Structure and functions of an Ecosystem, Food chains and Food webs, Ecological succession, Ecological pyramids, Biogeochemical cycles
- Biodiversity, Values of biodiversity, Bio geographical classification of India. Endangered and endemic species of India, Threats to biodiversity; Man and wild life conflicts, Conservation of biodiversity: In-situ and Ex-situ conservation methods

UNIT – IV : Environmental Pollution

Introduction to Environmental Pollution Causes, effects and control measures of:

Air pollution, Water pollution, Noise pollution, Solid Waste Management – Sources, Classification, effects and control measures of Municipal solid waste, Biomedical waste & Hazardous and e-waste, Disaster Management.

UNIT – V : Environmental Management

- Sustainable development and unsustainability
- Climate disruption, Greenhouse effect, Ozone layer depletion and Acid rain.
- Stockholm and Rio Summit
- Environmental Impact Assessment (EIA)
- Green building
- Environmental Law- Air, Water, Wildlife, Forest, and Environmental protection act

TEXTBOOKS:

1. Anubha Kaushik, C.P.Kaushik, “*Perspectives in Environmental Studies*”, 5th edition, New age international publishers, Delhi, 2016.
2. G. Tyler Miller, Scott Spoolman, “*Introduction to Environmental Studies*”, 13th Edition, Cengage Learning, New Delhi, 2009.

REFERENCE BOOKS:

1. M. Anji Reddy, “Textbook of Environmental Sciences and Technology”, 2nd Edition, BS Publications, Delhi 2011.
2. Deeshita Dave, P. Udaya Bhaskar, “Environmental Studies”, 2nd Edition, Cengage Learning, New Delhi, 2012.
3. S.Deswal, A. Deswal, “A Basic course in Environmental Studies”, 2nd Edition, Educational & Technical Publishers, Delhi, 2014.
4. R. Rajagopalan, “Environmental Studies (From Crisis to Cure)”, 3rd Edition, Oxford University Press, New Delhi, 2012.
5. De, A.K, “Environmental Chemistry”, 5th Edition, New Age International (P) Limited, New Delhi, 2003.
6. Dr.K.V.S.G. Murali Krishna, “Environmental Studies”, 1st Edition, VGS Techno Series, Vijayawada, 2010.
7. Mahua Basu, S.Xavier, “Fundamentals of Environmental Studies”, 1st edition, Cambridge University Press, Delhi, 2016.

B.Tech. (IV Sem.)

20CS58–DATA MINING USING PYTHON LAB

| L | T | P | Cr. |
|---|---|---|-----|
| 0 | 0 | 3 | 1.5 |

Pre-requisite : Python Programming.

Course Educational Objective: The objective of this lab is to Practical exposure on implementation of well-known data mining algorithms and Learning performance evaluation of data mining algorithms in a supervised and an un supervised setting.

Course Outcomes (CO): *At the end of this course, the student will be able to:*

- CO1:** Apply preprocessing techniques on real world datasets.(**Apply-L3**)
- CO2:** Apply apriori algorithm to generate frequent itemsets.(**Apply L3**)
- CO3:** Apply Classification and clustering algorithms on different datasets.(**Apply L3**)
- CO 4:** Improve individual / teamwork skills, communication & report writing skills with ethical values.

Use python library scikit-learn wherever necessary

1. Demonstrate the following data preprocessing tasks using python libraries.
 - a) Loading the dataset
 - b) Identifying the dependent and independent variables.
 - c) Dealing with missing data
2. Demonstrate the following data preprocessing tasks using python libraries.
 - a) Dealing with categorical data.
 - b) Scaling the features.
 - c) Splitting dataset into Training and Testing Sets
3. Demonstrate the following Similarity and Dissimilarity Measures using python
 - a) Pearson's Correlation
 - b) Cosine Similarity
 - c) Jaccard Similarity
 - d) Euclidean Distance
 - e) Manhattan Distance
4. Build a model using linear regression algorithm on any dataset.
5. Build a classification model using Decision Tree algorithm on iris dataset
6. Apply Naïve Bayes Classification algorithm on any dataset
7. Generate frequent itemsets using Apriori Algorithm in python and also generate association rules for any market basket data.
8. Apply K- Means clustering algorithm on any dataset.
9. Apply Hierarchical Clustering algorithm on any dataset.
10. Apply DBSCAN clustering algorithm on any dataset.

Web Resources:

1. <https://analyticsindiamag.com/data-pre-processing-in-python/>
2. <https://towardsdatascience.com/decision-tree-in-python-b433ae57fb93>
3. <https://towardsdatascience.com/calculate-similarity-the-most-relevant-metrics-in-a-nutshell-9a43564f533e>
4. <https://www.springboard.com/blog/data-mining-python-tutorial/>
5. <https://medium.com/analytics-vidhya/association-analysis-in-python-2b955d0180c>
6. <https://www.datacamp.com/community/tutorials/naive-bayes-scikit-learn>
7. <https://www.analyticsvidhya.com/blog/2019/05/beginners-guide-hierarchical-clustering/>
8. <https://towardsdatascience.com/dbscan-algorithm-complete-guide-and-application-with-python-scikit-learn-d690cbae4c5d>

B.Tech. (IV Sem.)

**20IT54 - OPERATING SYSTEMS and LINUX
INTERNALS LAB**

| L | T | P | Cr. |
|---|---|---|-----|
| 0 | 0 | 3 | 1.5 |

Pre-requisite : Programming language, DBMS, OOP concepts

Course Educational Objective: The objective of this lab is to provide the various UNIX/Linux operating system commands, importance of System calls, Scheduling algorithms and Memory Management techniques.

Course Outcomes (CO): *At the end of this course, the student will be able to:*

- CO1:** Experiment with Unix commands and shell programming (**Understand- L2**)
- CO2:** Implement CPU scheduling algorithms and memory management techniques (**Apply- L3**).
- CO3:** Simulate process synchronization and file system management using system calls (**Apply –L3**).
- CO 4:** Improve individual / teamwork skills, communication & report writing skills with ethical values.

Cycle 1: Learn some of the basic concepts in OS with the help of Linux commands. Commands: ps, kill, killall, ls, ln, readlink, cp, rm, vi editor, grep, find, who, cat, who.

Cycle 2: Introduce system calls in the Linux OS with the help of some basic system calls such as fork, exec, sleep, wait, etc.

Cycle 3: Write a shell script that accepts a file name, starting and ending line numbers as arguments and displays all the lines between the given line numbers.

Cycle 4: Write a shell script that deletes all lines containing a specified word in one or more files supplied as arguments to it.

Cycle 5: Write a Program that creates a Custom Shell.

Cycle 6: Learn about some other system calls in UNIX based systems such as Signals, Signal Handling, and Pipes.

Cycle 7: Write a program to simulate the following CPU scheduling algorithms to find turnaround time and waiting time.

- a) FCFS
- b) Round Robin

Cycle 8: Write a program to simulate paging technique of memory management.

Cycle 9: Write a program that takes one or more file/directory names as command line input and reports the following information on the file.

- a) File type.
- b) Number of links.
- c) Time of last access.
- d) Read, Write and Execute permissions.

Cycle 10: Write a Program that creates a Custom File System.

Online references:

1. <https://github.com/brenns10/lsh/tree/407938170e8b40d231781576e05282a41634848c>
2. <https://github.com/MaaSTaaR/SSFS>

B.Tech. (IV Sem.)

20IT55 - SOFTWARE ENGINEERING LAB

| L | T | P | Cr. |
|---|---|---|-----|
| 0 | 0 | 3 | 1.5 |

Pre-requisite : Object Oriented Programming

Course Educational Objective: The main objective of this course is that a student will be familiar with principles behind the Object-Oriented Design and able to apply those principles in a project setting. Students will analyze applications and know how to take a pragmatic approach to software design and development.

Course Outcomes (CO): *At the end of this course, the student will be able to:*

- CO1:** Analyze Software Requirements for the given Real-World Application using Use Cases.(**Analyze-L4**)
- CO2:** Develop the UML Diagrams to view Software System in static aspects. (**Analyze-L4**)
- CO3:** Develop the UML Diagrams to view Software System in dynamic aspects.(**Analyze-L4**)
- CO 4:** Improve individual / teamwork skills, communication & report writing skills with ethical values.

Consider the following Case Studies:

- 1) Automated Teller Machine (ATM)
- 2) Library Management System
- 3) Railway Ticket Reservation System
- 4) Point-of-Sale Terminal
- 5) Customer Support Service Operations
- 6) Cab Booking Service

Cycle-1: Analyze the Requirements for the following Case Studies.

- 1) Automated Teller Machine (ATM)
- 2) Library Management System
- 3) Railway Ticket Reservation System

Cycle-2: Analyze the Requirements for the following Case Studies.

- 1) Point-of-Sale Terminal
- 2) Customer Support Service Operations
- 3) Cab Booking Service

Cycle-3: Basics of UML

- 1) Introduction to UML.
- 2) Familiarization with any one of the Software such as Rational Rose or Umbrello or Gliffy Diagram etc.

Cycle-4: For each case study given earlier, Construct Use Case Diagram for following:

- 1) Identify and Analyze the Actors.
- 2) Identify the Actions.
- 3) Analyze the Relationships between Actors and Actions.
- 4) Sketch the Use Case Diagram.

Cycle-5 and Cycle-6:For each case study given earlier, Construct Class Diagram in the following manner:

- 1) Identify and Analyze the Classes related to your problem.
- 2) Analyze the Attributes and Operations
- 3) Analyze the Relationships between Classes
- 4) Sketch the Class Diagram

Cycle-7:For each case study given earlier, Construct Interaction Diagrams in the following manner:

- 1) Identify the Objects participating in Communication.
- 2) Identify the Messages between the objects.
- 3) Give numbering to messages.
- 4) Use Flat Sequencing or Procedural Sequencing for numbering.

Cycle-8:For each case study given earlier, Construct Activity Diagram in the following manner:

- 1) Identify activities in your case study.
- 2) Identify relationships among activities.
- 3) Use Fork or Join, if necessary.
- 4) Sketch the diagram.

Cycle-9:For each case study given earlier, Construct State Chart Diagram in the following manner:

- 1) Identify the different states in your case study.
- 2) List out the different sub-states present in the state.
- 3) Identify relationships among the state to state.
- 4) Sketch the diagram.

Cycle-10:For each case study given earlier, Construct Component Diagram in the following manner:

- 1) Identify the different components in your case study.

- 2) Create a visual for each of the component.
- 3) Describe the organization and relationships between components using interfaces, ports etc.
- 4) Sketch the diagram.

Cycle-11:For each case study given earlier, Construct Deployment Diagram in the following manner:

- 1) Identify the nodes.
- 2) Identify the relationships among the nodes.
- 3) Sketch the Diagram.

REFERENCE BOOK(S):

1. Roger S. Pressman, “Software engineering- A practitioner’s Approach”, TMH International Edition, 6th edition, 2005.
2. Grady Booch, James Rumbaugh, Ivar Jacobson, “The Unified Modeling Language UserGuide”, Pearson Education, 2nd Edition, ISBN: 0-201-57168-4, 1998.
3. Meilir Page-Jones, “Fundamentals of Object-Oriented Design in UML”, Pearson Education, 1st Edition, ISBN: 9788177586770, 8177586777, and 2007.
4. Robert B. Jackson, Stephen D. Burd, John W. Satzinger, “Object-Oriented Analysis and Design with the Unified Process”, Cengage Learning, 1st Edition, ISBN: 9788131502693, 8131502694, 2007.
5. Craig Larman, “Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development”, PHI, 3rd Edition, ISBN: 978-0131489066, 2004.

B.Tech. (IV Sem.) **20CSS2 – WEB APPLICATION DEVELOPMENT
USING FULL STACK – Module-II
(Backend Development)**

| L | T | P | Cr. |
|---|---|---|-----|
| 1 | 0 | 2 | 2 |

Pre-requisite : Object Oriented Programming and Data Base Management Systems.

Course Educational Objective:The objective of this course is to learn the importance of client-server architecture in the web application development and able to develop dynamic data driven web applications by using advanced java technologies (Servlets, JSP, Struts2 and Hibernate framework).

Course Outcomes (CO): *At the end of this course, the student will be able to:*

- CO1:** Apply JDBC concepts to establish the communication between Java Applications and database. **(Apply-L3)**
- CO2:** Develop Static and Dynamic Web Applications by using Servlets and Java Server Pages (JSP). **(Apply L3)**
- CO3:** Develop Dynamic Data Driven Web Applications by using Struts2 and Hibernate frameworks. **(Apply L3)**
- CO 4:** Improve individual / teamwork skills, communication & report writing skills with ethical values.

Module 1:

- a) Develop a java program to connect oracle database by using JDBC Connection string.
- b) Develop a java program to prepare the Statement to execute the query to insert the number of records into oracle database.
- c) Develop a java program to fetch the records from the oracle database table by using ResultSet Interface.

Module 2:

- a) Develop a java program to prevent the SQL Injection attacks by using PreparedStatement.
- b) Develop a java program to execute stored procedures and sub functions by using Callable Statement.
- c) Develop a java program to Scroll the ResultSet data in both forward and backward directions by using ScrollableResultSet.
- d) Develop a java program to print the database table meta data by using ResultSetMetaData.

Module 3:

- a) Develop a Servlet program to demonstrate the role of Servlet Life Cycle methods in the web application.
- b) Develop a Servlet program to access Init parameter values from web.xml to Servlet program by using ServletConfig interface.
- c) Develop a Servlet program to navigate from one Servlet page to another Servlet page by using RequestDispatcher interface.

Module 4:

- a) Develop a Servlet program to AUTHENTICATE User details from oracle database by using JDBC connectivity. (Dynamic Login Checking for the registered users in the oracle database)
- b) Develop a Servlet program to implement Session Management concept by using HttpSession.

Module 5:

- a) Develop a JSP program to demonstrate the importance of Scripting elements in JSP.
- b) Develop a JSP program to illustrate the importance of implicit objects in JSP.
- c) Develop a JSP program to access JavaBean class by using jsp:useBean, jsp:setProperty and jsp:getProperty.

Module 6:

- a) Develop a JSP program to insert and retrieve the records from the oracle database.
- b) Develop a JSP program to demonstrate the session management in between login and logout period of the end user.
- c) Develop a JSP program to illustrate the process of handling errors and exceptions in JSP pages.

Module 7:

- a) Develop a Struts2 web application to perform Insert, Update and Delete operations in the oracle database table through JDBC.
- b) Develop a Struts2 web application to fetch all the records of the table created in the above from oracle database.

Module 8:

- a) Develop a Program how to code a basic Java Hibernate program by using eclipse IDE.
- b) Develop a web application to insert the record of the user in the oracle database by using Hibernate.

REFERENCE BOOK(S):

1. Herbert Schildt, “Java: The complete reference”, TMH Publications, 7th edition, 2006.
2. Kathy Sierra & Bert Bates, “Head First Servlets and JSP: Passing the Sun Certified Web Component Developer Exam”, O'Reilly Publications Second Edition.
3. Budi Kurniawan, “Struts 2 Design and Programming: A Tutorial”, BrainySoftware, 2nd Edition, 2008.
4. Christian Bauer, Gavin King, Gary Gregory “Java Persistence with Hibernate: Revised Edition of Hibernate in Action Paperback”, Manning Publication, 2nd Edition, 2006.
5. Santosh Kumar K, “JDBC 4.2, Servlet 3.1, and JSP 2.3 Includes JSF 2.2 and Design Patterns, Black Book”, Dreamtech publication, 2nd Edition.
6. Mahmoud Parsian, “JDBC Recipes: A Problem-Solution Approach”, Apress.
7. Madhu sudhan Konda, “Just Hibernate, A Lightweight Introduction to the Hibernate Framework”, O'Reilly Media.
8. Chuck Cavaness, “Programming Jakarta Struts”, O'Reilly Media, 2nd Edition.