**NATIONAL INSTITUTE OF ELECTRONICS AND INFORMATION TECHNOLOGY**

DEEMED TO BE UNIVERSITY UNDER DISTINCT CATEGORY

(AN AUTONOMOUS INSTITUTION UNDER MINISTRY OF ELECTRONICS & IT, GOVT. OF INDIA)

# Curriculum and Syllabi for

# Diploma - Computer Science and Technology

## (2025-26 Scheme)

रा.इ.सू.प्रौ.सं

NIELIT

Deemed to be University

Main Campus at Ropar and Constituent Units at Aizawl, Agartala, Aurangabad, Calicut, Gorakhpur, Imphal, Itanagar, Kekri, Kohima, Patna and Srinagar

## Vision

The Diploma in Computer Science and Technology at NIELIT Deemed To Be University aims to be a center of excellence in computer science education that nurtures technically skilled, innovative, and ethical professionals who contribute to societal and industrial advancement in the digital era.

## Mission

The 3-year Diploma in Computer Science and Technology aims to provide a strong foundation in computer science through a balanced blend of theory and practical skills, fostering innovation, problem-solving ability, and lifelong learning. The program aims to produce competent, socially responsible, and industry-ready professionals who can adapt to the evolving needs of technology and society.

## Program Education Objectives

PEO 1: Candidate will apply fundamental knowledge of computer science, programming, and engineering principles to solve real-world technical problems in industry and society.
PEO 2: Candidates will pursue higher education or engage in continuous learning to enhance their knowledge and skills in emerging technologies.
PEO 3: Candidates will practice their profession with integrity, respect for diversity, and awareness of ethical, social, and environmental responsibilities.
PEO 4: Candidates will engage in continuous learning to keep pace with rapidly changing technologies and industry trends.

## Program Outcomes

1. Apply knowledge of mathematics, science, engineering fundamentals, and computer science concepts to solve computing problems.
2. Identify, analyze, and formulate solutions for well-defined computing problems using appropriate techniques.
3. Design, develop, test, and maintain computer-based systems to meet desired requirements within realistic constraints.
4. Communicate effectively on technical activities with the computing community and society at large, including writing reports, making presentations, and giving clear instructions.
5. Function effectively as an individual and as a member or leader in diverse teams and multidisciplinary settings.

## Program Specific Outcomes

PSO 1: Candidates will be able to design, develop, test, and maintain reliable and efficient software systems using modern programming languages, tools, and technologies.
PSO 2: Candidates will be able to install, configure, and troubleshoot computer networks and hardware components to meet organizational and societal needs.
PSO 3: Candidates will be able to apply knowledge of emerging technologies such as cloud computing, IoT, AI, and cybersecurity to solve real-world problems and innovate solutions.

## CATEGORY WISE CREDIT DISTRIBUTION

| Category | Category Code | Credits |
|---|---|---|
| Skill / Employment Enhancement Courses (Project/Internship/Seminar/Dissertation/Research) | EE | 20 |
| Value Added Courses | VA | 4 |
| Ability Enhancement Courses | AE | 4 |
| Audit Courses (without grade or credit) | AU | 0 |
| Open Elective Courses | OE | 8 |
| Professional Elective Courses | PE | 16 |
| Program Core Courses | PC | 72 |
| **Total Credits (Common track)** | | **124** |

## EVALUATION AND ASSESSMENT

1. Performance of a student in a semester shall be evaluated through continuous Class assessment, Tutorial/Lab assessment, Mid-Semester Examination (MSE) and End-Semester Examination (ESE). Both the MSE and ESE shall be the University examination and will be conducted as notified by the Controller of Examinations (CoE) of the University.

2. The continuous assessment shall be based on assignments, tutorials, paper presentation / quizzes/ viva-voce / flipped classes, lab work / projects / fieldwork and attendance, etc.

3. The MSE/ESE shall be comprising of written papers.

4. The overall assessment of the students will be done as per the following scheme:

| S. No. | Assessment Type | Marks Weightage |
|---|---|---|
| 1. | Mid Semester Examination | 20 |
| 2. | End Semester Examination | 40 |
| 3. | Continuous Assessment - Practical /Lab/ Tutorial | 25 |
| 4. | Continuous Assessment - Theory | 15 |
| **Total** | | **100** |

For more details, please refer the applicable ordinance for this programme and Assessment SOPs.

# CURRICULUM

| Semester 1 | | | | | | |
|---|---|---|---|---|---|---|
| **Semester Code** | **Course Code** | **Course Title** | **L** | **T** | **P** | **Cr** |
| PC-DIPCST-101 | DASH250006 | Applied Mathematics-I | 3 | 1 | 0 | 4 |
| PC-DIPCST-102 | DASH250005 | Applied Science | 3 | 0 | 2 | 4 |
| PC-DIPCST-103 | DCSE250032 | Basic Computer Science and Electronics | 2 | 0 | 4 | 4 |
| PE-DIPCST-104 | Elective | Program Elective | | | | 4 |
| OE-DIPCST-105 | Elective | Open Elective | | | | 4 |
| AU-DIPCST-106 | Elective | Audit Elective | | | | 0 |

| Semester 2 | | | | | | |
|---|---|---|---|---|---|---|
| **Semester Code** | **Course Code** | **Course Title** | **L** | **T** | **P** | **Cr** |
| PC-DIPCST-201 | DASH250007 | Applied Mathematics-II | 3 | 1 | 0 | 4 |
| PC-DIPCST-202 | DCSA250010 | Technical Drawing using CAD | 2 | 0 | 4 | 4 |
| PC-DIPCST-203 | DCSE250036 | Basic Programming Concepts | 2 | 0 | 4 | 4 |
| AE-DIPCST-204 | DASH250016 | Communication Skills | 1 | 0 | 2 | 2 |
| VA-DIPCST-205 | DASH250101 | Sports and Yoga | 0 | 0 | 4 | 2 |
| EE-DIPCST-206 | EE | Internship-I | | | | 4 |

| Semester 3 | | | | | | |
|---|---|---|---|---|---|---|
| **Semester Code** | **Course Code** | **Course Title** | **L** | **T** | **P** | **Cr** |
| PC-DIPCST-301 | DCSE250030 | Fundamentals of Computer Organization and Architecture | 2 | 0 | 4 | 4 |
| PC-DIPCST-302 | DCSA250057 | Object Oriented Programming using C++ | 3 | 0 | 2 | 4 |
| PC-DIPCST-303 | DCSE250056 | Data Structures and Algorithms | 3 | 0 | 2 | 4 |
| PE-DIPCST-304 | Elective | Program Elective | | | | 4 |
| OE-DIPCST-305 | Elective | Open Elective | | | | 4 |
| AU-DIPCST-306 | Elective | Audit Elective | | | | 0 |

| Semester 4 | | | | | | |
|---|---|---|---|---|---|---|
| **Semester Code** | **Course Code** | **Course Title** | **L** | **T** | **P** | **Cr** |
| PC-DIPCST-401 | DCSE250102 | Operating Systems | 3 | 0 | 2 | 4 |
| PC-DIPCST-402 | DCSE250060 | Database Management Systems | 2 | 0 | 4 | 4 |
| PC-DIPCST-403 | DCSE250035 | Computer Networks | 3 | 0 | 2 | 4 |
| PE-DIPCST-404 | Elective | Program Elective | | | | 4 |
| AE-DIPCST-405 | DASH250091 | Professional Values and Ethics | 2 | 0 | 0 | 2 |
| VA-DIPCST-406 | DASH250052 | Environmental Science | 2 | 0 | 0 | 2 |
| EE-DIPCST-407 | EE | Minor Project | | | | 4 |

| Semester 5 | | | | | | |
|---|---|---|---|---|---|---|
| **Semester Code** | **Course Code** | **Course Title** | **L** | **T** | **P** | **Cr** |
| PC-DIPCST-501 | DCSE250079 | Fundamentals of Programming in Python | 2 | 0 | 4 | 4 |
| PC-DIPCST-502 | DCSE250120 | Theory of Computation | 3 | 1 | 0 | 4 |
| PC-DIPCST-503 | DCSE250088 | Internet and Web Technologies | 3 | 0 | 2 | 4 |
| PC-DIPCST-504 | DCSA250071 | Software Engineering | 3 | 1 | 0 | 4 |
| PE-DIPCST-505 | Elective | Program Elective | | | | 4 |
| EE-DIPCST-506 | EE | Summer Internship-II | | | | 2 |
| EE-DIPCST-507 | EE | Major Project - I | | | | 2 |

| Semester 6 | | | | | | |
|---|---|---|---|---|---|---|
| **Semester Code** | **Course Code** | **Course Title** | **L** | **T** | **P** | **Cr** |
| PC-DIPCST-601 | DOAI250013 | Machine Learning Fundamentals | 2 | 0 | 4 | 4 |
| PC-DIPCST-602 | DCSA250003 | Mobile Application Development using Android | 2 | 1 | 2 | 4 |
| AU-DIPCST-603 | Elective | Audit Elective | | | | 0 |
| EE-DIPCST-604 | DASH250050 | Entrepreneurship and Start-Ups | 3 | 1 | 0 | 4 |
| EE-DIPCST-605 | EE | Major Project -II | | | | 2 |
| EE-DIPCST-606 | EE | Technical Seminar | | | | 2 |

# Elective Courses #

| Elective Courses for PE Category | | | | | | |
|---|---|---|---|---|---|---|
| **Course Code** | **Course Title** | **L** | **T** | **P** | **Cr** | **For Sem./Code** |
| DCSA250039 | Introduction to IT Systems | 2 | 1 | 2 | 4 | 104 |
| DCSE250078 | Fundamentals of ICT | 3 | 0 | 2 | 4 | 104 |
| DCSA250001 | Fundamentals of Image and Video Processing | 2 | 0 | 4 | 4 | 304 |
| DCSE250029 | Computer Graphics | 3 | 1 | 0 | 4 | 304 |
| DCSA250065 | Programming in Java | 3 | 1 | 0 | 4 | 404 |
| DOAI250009 | Big Data Analytics | 3 | 0 | 2 | 4 | 404 |
| DCSA250055 | Multimedia Technologies | 3 | 0 | 2 | 4 | 505 |
| DCSE250020 | Cloud Computing | 3 | 0 | 2 | 4 | 505 |

| Elective Courses for OE Category |
|---|
| Students may opt courses under MOOC, NPTEL, SWAYAM, NSQF/NCVET aligned courses or other relevant technical subjects not included in their core curriculum. The exercise of option shall be subject to the Course Schedule of the respective Constituent Unit, the credit requirements and availability of seats. Guidelines for choosing MOOC/Online courses are given separately and shall have to be adhered to. |

| Elective Courses for AU (Audit) Category | | | | | |
|---|---|---|---|---|---|
| **Course Code** | **Course Title** | **L** | **T** | **P** | **Cr** |
| DASH240027 | Disaster Management | 2 | 0 | 0 | 0 |
| DASH240047 | English for Research Paper Writing | 2 | 0 | 0 | 0 |
| DASH240052 | Environmental Science | 3 | 0 | 0 | 0 |
| DASH240085 | Pedagogy Studies | 2 | 0 | 0 | 0 |
| DASH240086 | Personality Development through Life Enlightenment Skills | 2 | 0 | 0 | 0 |
| DASH240097 | Sanskrit for Technical Knowledge | 2 | 0 | 0 | 0 |
| DASH240103 | Stress Management by Yoga | 2 | 0 | 0 | 0 |
| DASH240104 | Technical Presentation and Report Writing | 0 | 0 | 2 | 0 |
| DASH240106 | Value Education | 2 | 0 | 0 | 0 |
| DASH240124 | Constitution of India - AU | 2 | 0 | 0 | 0 |
| DASH250023 | Constitution of India | 2 | 1 | 0 | 0 |
| DASH250027 | Disaster Management | 3 | 0 | 0 | 0 |
| DASH250034 | Energy and Environmental Engineering | 3 | 0 | 0 | 0 |
| DASH250047 | English for Research Paper Writing | 2 | 0 | 0 | 0 |
| DASH250054 | Essence of Indian Knowledge and Tradition | 2 | 0 | 0 | 0 |
| DASH250085 | Pedagogy Studies | 2 | 0 | 0 | 0 |
| DASH250086 | Personality Development | 2 | 0 | 0 | 0 |
| DASH250097 | Sanskrit for Technical Knowledge | 3 | 0 | 0 | 0 |
| DASH250103 | Stress Management by Yoga | 3 | 0 | 0 | 0 |
| DASH250106 | Value Education | 2 | 0 | 0 | 0 |

| | | | | | |
|---|---|---|---|---|---|
| DASH250117 | Innovative Thinking and Entrepreneurship Skills | 1 | 0 | 0 | 0 |
| DASH250118 | Intellectual Property Rights | 2 | 0 | 0 | 0 |
| DCSA240080 | Training on Computer Networking | 0 | 0 | 2 | 0 |
| DCSA240304 | Lab Based on Statistical Package | 0 | 0 | 2 | 0 |
| Any other Scheduled Course as per the Course Schedule of the respective Constituent Unit can also be chosen, subject to availability of seats. However, there shall be no assessment and grading for the course chosen as Audit Course. | | | | | |

# The choice of all type of Electives available shall be as per the Course Schedule of the respective Constituent Unit.

## Guidelines for Massive Open Online Courses (MOOC) / approved Online Courses

1. Relevant Swayam, MOOC, NPTEL, NIELIT NSQF and any other online courses approved by UGC/AICTE shall also be allowed as Open Electives(OE).

2. One credit of MOOC course should be minimum of 30 hours.

3. A student can choose any approved online course as Open Elective, subject to minimum credit requirements.

4. The students willing to opt OE under MOOC in their next semester, will submit their request to the MOOC Coordinator at their respective campus.

5. Once approved, the campus shall display the list of accepted courses.

6. The student has to submit certificate issued by the institution offering the MOOCs course, along with the number of credits and grades, to get credits transferred into his/her marks certificate issued by NDU.

7. The transfer of credits shall be as adopted by NDU.

## Detailed Syllabus

| Course Code | Course Name | L | T | P | Cr |
|---|---|---|---|---|---|
| DASH250006 | Applied Mathematics-I | 3 | 1 | 0 | 4 |

### Course Outcomes:

CO1: Perform vector operations (dot, cross, scalar triple product) and apply them in simple engineering problems.
CO2: Calculate derivatives of various functions and apply them in real-life rate and change-related problems.
CO3: Solve problems involving complex numbers using algebraic methods and De Moivre's Theorem.
CO4: Evaluate and apply trigonometric and inverse trigonometric identities to triangles and angle-based problems.
CO5: Use combinatorial methods and binomial expansions to solve mathematical and logical problems in computing contexts.

### Module 1:

Vector Algebra (9 hrs)

Definition of vector and scalar quantities, addition and subtraction of vectors, Dot and Cross product, Scaler and vector triple product, Angle between two vectors.

### Module 2:

Differential Calculus (9 hrs)

Functions and Their Graphs: Definition and types of functions, Graphical representation of functions, Domain and range of functions (elementary cases)
Limits and Continuity: Concept of limit of a function, Left-hand limit (LHL) and Right-hand limit (RHL), Standard limit formulas, Elementary methods for finding limits, Continuity of functions at a point and over an interval, Basic test for continuity and differentiability
Differentiation Techniques: Definition of derivative, Basic rules of differentiation: Sum and difference, Product rule, Quotient rule, Chain rule (function of a function)
Derivatives of Trigonometric functions, Exponential functions, Logarithmic functions, Inverse trigonometric functions.

### Module 3:

Complex Numbers and Their Applications ( 8 hrs)

Complex Numbers: Definition and representation of complex numbers (Cartesian form), Real and imaginary parts
Basic Operations: Addition, subtraction, multiplication, division, modulus, and conjugate
De Moivre's theorem: Statement and simple applications for powers and roots.

### Module 4:

Trigonometry (10 hrs)

Revision: Measurement of an angle (degree and radian), Relation Between degree and radian, Trigonometric ratios, Fundamental identities
Inverse trigonometric ratios: Definition of inverse trigonometric ratios, Principal values of Inverse trigonometric ratios, Relation between inverse trigonometric ratios
Properties of Triangle: Sine, Cosine, Projection and tangent rules (without proof), Simple problems.

### Module 5:

Permutations, Combinations & Binomial Theorem (9 hrs)

Permutations and Combinations: Factorial notation and basic rules, Permutation and Combination formulas, Application in real-life and computing problems
Binomial Theorem: Expansion for positive integral index (without proof), Expansion for negative index (expansion only).

### Labs / Practicals:

N/A

### References:

1. B.S. Grewal, Higher Engineering Mathematics, Khanna Publishers
2. N.P. Bali and Manish Goyal, A Textbook of Engineering Mathematics (Vol. I & II), Laxmi Publications
3. H.K. Dass, Polytechnic Mathematics, S. Chand Publishing
4. S.S. Sabharwal, Diploma Engineering Mathematics, Unique Publishers
5. R.D. Sharma, Mathematics for Class 11 & 12, Dhanpat Rai Publications
6. M.K. Jain or B.V. Ramana, Engineering Mathematics (Vol. I), S. Chand Publishing / Pearson

| Course Code | Course Name | L | T | P | Cr |
|---|---|---|---|---|---|
| DASH250005 | Applied Science | 3 | 0 | 2 | 4 |

## Course Outcomes:

CO1 : Demonstrate understanding of basic measurement principles and physical quantities.
CO2 : Apply laws of mechanics, heat, and material properties in solving engineering problems.
CO3 : Explain atomic structure, electrochemical processes, and corrosion.
CO4 : Identify and classify engineering materials including polymers and alloys.
CO5 : Perform fundamental experiments in Physics and Chemistry relevant to electronics and communication engineering.

## Module 1:

Unit, Dimension, Measurement and Mechanics
Concept of unit of physical parameters, Fundamental and derived units, S.I. system of units of different physical parameters, Dimension with examples of different physical parameters, measuring devices.
Concept of scalar and vector quantities, Speed, velocity and linear acceleration, Force, inertia, momentum, Conservation of linear momentum, Work, Power and Energy, Rotational Motion.

## Module 2:

Properties of matter: Physical and Chemical
Basic crystal structure, States of matter, elasticity, Expansions of solids, liquids and gases, Co-efficient of linear, surface and cubical expansion and relationship amongst them, Simple problems.
Heat and Temperature: Concept and Modes of heat transfer (Conduction, convection and Radiation with examples), Scales of temperatures and their relationship, thermometers and their uses.

## Module 3:

Atomic Structure, Chemical Bonding and Solutions
Rutherford, Bohr's atomic model, Pauli's exclusion principle, hydrogen spectrum, periodic table, Planck's constant and Heisenberg uncertainty principle, Quantum numbers, concept of s, p, and d orbitals. Concept of chemical bonding: cause of chemical bonding. Types of bonds: covalent bond, ionic bond and metallic bond, Solution: idea of solute, solvent and solution, methods to express the concentration of solution-molarity (M, molar constant), ppm, mass percentage, volume percentage and mole fraction.

## Module 4:

Electrolysis
Process of electrolysis: Equivalent weight and Faraday's laws of electrolysis, ionization, oxidation, reduction, electroplating, Electrolytic refining, Electrometallurgy etc.
Water and its classification, pH value and corrosive effects, total dissolved substance (TDS) acidity and alkalinity estimation.
Chemistry of Electrolytic cells, both during charging and discharging of batteries (Lithium ion, Lithium polymer, lead acid and NMH).

## Module 5:

Engineering Materials
Natural occurrence of metals: minerals, ores of iron, aluminum and copper, gangue (matrix), flux, slag, metallurgy: brief account of general principles of metallurgy, Extraction of iron from haematite ore using blast furnace, aluminum from bauxite along with reactions, silicon from silica (sand). Alloys and mixtures: definition, purposes of alloying, ferrous alloys and non-ferrous with suitable examples, properties and applications. General chemical composition, composition-based applications: Polymers: elastomers, thermos plastics and thermosetting polymers, composition and properties of ceramic, PVC, PTFE, Nylon and Bakelite.

## Labs / Practicals:

1. Use of Vernier Caliper to measure length, diameter, and depth.
2. Use of Screw Gauge and Spherometer to measure thickness and radius.
3. Determine Young's modulus by Searle's apparatus.
4. Determine surface tension by drop count method.
5. Verify laws of motion using inclined plane or pulley apparatus.
6. Determine coefficient of viscosity by using a viscometer.
7. Determine coefficient of linear expansion using brass or copper rod.
8. Compare specific heats of solids using method of mixtures.
9. Volumetric analysis using acid-base titration (standardization method).
10. Determination of water hardness using EDTA method.
11. Estimation of chloride content in water.
12. Determination of iron content using potassium dichromate.
13. Determination of conductivity of given solution.
14. Preparation of standard solution (Na2CO3 or oxalic acid).
15. Measurement of pH using pH meter or indicator strips.
16. Study of electrochemical cell and verification of Faraday's laws.

### References:

1. S.L. Gupta & Sanjeev Gupta, "Engineering Physics", Dhanpat Rai Publications.
2. Engineering Physics by PV Naik, Pearson Education Pvt. Ltd, New Delhi.
3. Jain & Jain, "Engineering Chemistry", Dhanpat Rai Publications.
4. Dr. Vairam, S., Engineering Chemistry, Wiley India Pvt.Ltd., New Delhi.

| Course Code | Course Name | L | T | P | Cr |
|---|---|---|---|---|---|
| DCSE250032 | Basic Computer Science and Electronics | 2 | 0 | 4 | 4 |

### Course Outcomes:

CO1. Use relevant diode in electronics circuits.
CO2. Use BJT in electronics circuits.
CO3. Use of BJT as amplifier and switch.
CO4. Use FET and MOSFET in electronics circuits.
CO5. Maintain DC regulated power supply.
CO6. Understand basic computer hardware and software components.
CO7. Operate common operating systems and productivity tools.
CO8. Apply fundamental cyber safety principles and protect digital systems.

### Module 1:

Applications of Diode
Different types of diodes and their materials: Construction, Symbol, working principle, applications, Forward and reverse biasing and V-I characteristics of following diodes: P-N junction diode, Zener diode, LED, Photo diode, Schottky diode, Diode as rectifier: Types of Rectifiers, Half wave, Full wave (bridge rectifier and center tapped), circuit operation, Input- output waveform for voltage and current, Parameters of rectifier: Average DC value, value of current and voltage, ripple factor, ripple frequency, PIV of diode, TUF, efficiency of rectifier. Types of Filters: Shunt capacitor, Series inductor, LC and CLC filter. Rectifier IC – KBU 808 IC pin diagram and application .

### Module 2:

Bipolar Junction Transistor
Current operating device, Different types of transistors: PNP, NPN, Transistor configurations: CB, CE, CC Transistor characteristics (input, and output) in different transistor configuration. Relation between alpha ,beta, gama. Comparison between CB, CC and C, BJT biasing: Need of DC load Line, Operating point, stabilization, thermal runaway, heat sink. Types of biasing: fixed biasing, base bias with emitter feedback, voltage divider. Classification of amplifier, BJT as an amplifier. Single Stage Amplifier: Working , various currents (Ib, Ic,Ie), Voltage gain of CE amplifier (no derivations required)

### Module 3:

Introduction to Computers and Internet:
Basic components: CPU, Memory (types), Display, HDD, SSD, peripherals, Software types, Internet basics: browsers, search engines, types of websites, Digital India portals awareness

### Module 4:

Operating Systems, Office Tools, and Information Security
Types and definitions of OS, Windows and Linux installation and procedures, Introduction to MS Office (Word, Excel, PowerPoint) and G-Suite (Drive, Docs, Sheets, Slides, Forms), Basics of cyber security: malware, viruses, anti-virus, firewalls, Best practices for system safety and data privacy

### Module 5:

Regulators and Power supply
Need of Regulated power supply . Basic block diagram of DC regulated power supply and function of each block,Load and Line regulation, Zener diode voltage regulator, Fixed voltage IC Regulator: Three terminal Pin diagram, working and application of 78XX and 79xx series, Variable voltage IC Regulator : IC 723 pin diagram , block diagram, working. Low voltage regulator, High voltage regulator, Switch Mode Power supply : Need of SMPS , block diagram and functions of blocks.

### Labs / Practicals:

1. Test the performance of PN Junction diode
2. Test the performance of Zener diode
3. Construct and test the Half-Wave Rectifier
4. Construct and test the Full-Wave Rectifier
5. Test and plot input/output characteristics of BJT
6. BJT as a switch and amplifier
7. Assembling and disassembling of computer components
8. Explore features of peripheral devices and install drivers
9. Create email, use browsers, search engines, and cloud storage
10. Browse and report on Digital India portals
11. Install Windows and Linux OS in lab systems
12. Create files using MS Word, Excel, PowerPoint, and G-Suite
13. Install and update antivirus software
14. Demonstrate use of firewall and report common security threats

### References:

1.Mehta, V.K. Mehta, Rohit Mehta, Principles of Electronics, S.Chand New Delhi, edition-2008 ISBN-13: 978-8121927833
2. P.Ramesh Babu Electronics Device and Circuits, Scitech Publications (India) Pvt Ltd ,ISBN-13 978-8183712156
3. ITL Education Solutions, Introduction to Computer Science, Pearson Education
4. R.S. Salaria, Computer Fundamentals, Khanna Publishing House

| Course Code | Course Name | L | T | P | Cr |
|---|---|---|---|---|---|
| DASH250007 | Applied Mathematics-II | 3 | 1 | 0 | 4 |

### Course Outcomes:

CO1 : Solve systems of equations using determinants and matrix algebra, including Cramer's rule and inverse matrix methods.
CO2: Evaluate definite and indefinite integrals using standard techniques and apply them to solve area and volume problems.
CO3: Calculate statistical measures such as mean, median, mode, variance, and standard deviation, and compare datasets effectively.
CO4: Apply concepts of differentiation to solve problems involving rates of change, tangents, normals, and optimization in technical scenarios.
CO5: Form and solve basic first-order differential equations using standard methods such as variable separation, exact, and linear types.

### Module 1:

Determinants and Matrices ( 9 hrs )
Matrices: Algebra of matrices; inverse of a matrix; solving a system of three linear equations using the matrix inverse method (for non-singular coefficient matrices).
Determinants: Definition and elementary properties of determinants of matrices up to the 3rd order; consistency of linear equations; Cramer's Rule.

### Module 2:

Integral Calculus ( 9 hrs )
Integration as the inverse operation of differentiation; simple integration by substitution, by parts, and by partial fractions (for linear factors only); use of standard formulas for integration of functions of the form x^m (a + b x^n), where m and n are positive integers.
Applications of integration: area bounded by a curve and coordinate axes; volume of a solid formed by the revolution of an area about the coordinate axes.

### Module 3:

Statistics ( 8 hrs )
Measures of central tendency, Mean, median, Mode
Measure of Dispersion mean, Standard deviation, Mean deviation, variance and coefficient of variation comparison of Two sets.

### Module 4:

Differential Calculus II ( 9 hrs )
Higher-order derivatives: concept and computation of second, third, and higher-order derivatives; Leibniz's theorem (statement only, no proof) and simple applications for finding the nth derivatives of basic functions.
Applications of derivatives:
-Geometrical: equation of a tangent and normal to a curve at a point; angle between two curves; finding points of inflection (basic idea only).
-Increasing/decreasing functions: determination of intervals of monotonicity using the first derivative test; finding local maxima and minima using first and second derivative tests.
-Rate of change and motion: derivative as a rate of change; velocity and acceleration in one-dimensional motion.

### Module 5:

Differential Equations ( 10 hrs )

Introduction, Formation of differential equations, Solution of differential equations, Separation of variables, Homogeneous equations, Exact Differential Equations, Integrating factor method, Linear differential equation.

### Labs / Practicals:

N/A

### References:

1. B.S. Grewal, Higher Engineering Mathematics, Khanna Publishers.
2. H.K. Dass & Er. Rajnish Verma, Mathematical Engineering, S. Chand Publications.
3. Erwin Kreyszig, Advanced Engineering Mathematics, Wiley India (Indian Adapted Edition).
4. R.K. Jain & S.R.K. Iyengar, Advanced Engineering Mathematics, Narosa Publishing House.
5. S.C. Gupta & V.K. Kapoor, Fundamentals of Mathematical Statistics, Sultan Chand & Sons.
6. P. Kandasamy, K. Thilagavathy, K. Gunavathi, Engineering Mathematics (Volumes I & II), S. Chand Publications

| Course Code | Course Name | L | T | P | Cr |
|---|---|---|---|---|---|
| DCSA250010 | Technical Drawing using CAD | 2 | 0 | 4 | 4 |

### Course Outcomes:

CO1. Explain the basics of AutoCAD software required for building drawing,
CO2. Comprehend the various components of building,
CO3. Design and draft the plan and elevation of a building using AutoCAD,
CO4. Design and draft the plan and sectional views of foundation and columns using AutoCAD
CO5. Design and draft the plan and sectional details of plinth and lintel beam using AutoCAD,
CO6. Develop and design the sectional views of a building using AutoCAD, and
CO7. Develop and design the plan and sectional view of a staircase using AutoCAD

### Module 1:

Fundamentals of CAD:
Introduction- CAD/CAM defined- the product cycle and CAD /CAM - automation and CAD/CAM- Computer Technology: Introduction-Central Processing unit-Types of memory-input output-data representation-computer programming languages-operating the computer system
CAD – Concept and Need, Functional Areas of CAD, Specifications of CAD work station, Input- Output devices, Features of CAD software.

### Module 2:

Production Drawing
AutoCAD and AutoCAD Mechanical, Requirement of production drawings, Content library, Construction features, Use of layers and Annotations

### Module 3:

Parametric Modeling:
Parametric and non-parametric modelling, 2D sketch environment of parametric software and non-parametric software, Parametric software in industries and their features, Sketching and solid modelling tools

### Module 4:

Advance Modeling techniques:
Advance tool and basic solid modelling tools. Various advance modelling tools (sweep, loft-blend, helical sweep, pattern, boundary, Standard holes wizard), Error rectification.

### Module 5:

Assembly Modeling and Drawing:
Basic Assembly constraints, Mechanical Assembly constraints, 2D production drawing using drawing modules with various orthographic views (may have sectional views), Generation of Bill of Material with balloons, plotting of assembly and drawing details on standard paper with title block mentioning scale of plot.

### Labs / Practicals:

Name of the experiment: -
1. Demonstration of AutoCAD Mechanical environment and its commands.
2. Prepare 2D production drawing and plot on paper using AutoCAD Mechanical (At least Four parts)
3. Demonstration of Parametric software environment and its commands.

4. Prepare 2D drawing using parametric software (At least Four parts)
5. Prepare 3D Solid Models using parametric softwares (includes base features like extrude, revolve, cut, shell, chamfer, rib, fillet etc.) (At least Eight parts)
6. Prepare 3D Solid Models using parametric softwares (Includes advance features like sweep, blend (loft), pattern
7. Prepare 2D production drawing of given components prepared in exercise 5 and 6 and take print using plotter. (at least four components)
8. Prepare 3D solid modeling of given Assembly and animate. (must 4 have 4 to 6 components) (at least four assemblies)

## References:

1. Tickoo, Sham Creo 2.0 for designer and engineers Dreamtech press, New Delhi
2. Rider, Michel J Designing with Creo Parametric 2.0 SDC Publications, Kansas, USA
3. Tickoo, Sham AutoCAD for engineers and Designers Dreamtech press, New Delhi
4. Narayan, K.L. Production drawing New Age publication, New Delhi

| Course Code | Course Name | L | T | P | Cr |
|---|---|---|---|---|---|
| DCSE250036 | Basic Programming Concepts | 2 | 0 | 4 | 4 |

### Course Outcomes:

CO1.Develop script to demonstrate use of basic building blocks of python.
CO2. Implement conditional and looping statements for given problem statement.
CO3. Perform operations on sequence structures in python.
CO4. Implement basics of object oriented programming concepts. Create modules and packages for given purpose.
CO5. Identify and use the syntax rules for constructing programmatic solutions in C.

### Module 1:

Introduction to C: Computing Devices, How Computer Stores Data, How Program Works Introduction to 'C' Language: features; applications of C; IDE; Compiler; C Program Structure, Execution of C programs, debugging C code, Data Types & Input/Output: Character set, Variables and Identifiers, Built-in Data Types, Variable Definition, Declaration, C Key Words-Rules & Guidelines for Naming Variables. Constants and Literals. Basic input/output statement (printf() and scanf()

### Module 2:

Operators and Expressions: Operators and Arithmetic, Conditions, Relational Operators, Logical Operators. Expressions, Precedence, and Order of Evaluation. Simple assignment statement, Decision Structure and Boolean Logic: if statement, if-else statement, nested if-else, ifelse ladder, Switch, Break, Continue, Arrays: What is an Array? Declaring an Array, Initializing an Array. One dimensional array: Array manipulation; Finding the largest/smallest element in array; Searching & Sorting of element from an array.

### Module 3:

Functions: Definition of Function, External Variables, Scope Rules, Header Files, Static Variables, Register Variables, Structures: Basic of Structures, Structures variables, initialization, structure assignment, nested structure, structures and arrays: arrays of structures, Union.

### Module 4:

Pre-processors - Introduction, Macro substitution, File inclusion. Includes directives like #include, #define , File Input/Output: File Operations, opening a File, File Opening Modes, reading from a File, writing to a File, Closing the File.

### Module 5:

Basic Python's Constructs
Introduction to Python- Python as scripting Language, Programming language Vs Scripting Language (C vs Python), Python's Technical Strength, Application in different domains, Python's building blocks- Identifiers, Keywords, Variables, Constants, Indentation, Comments in python, Python's Data Types – Numbers, Strings, List, Tuples, Dictionaries, Sets, Input and Output statements in python, Operators in Python- Operators as Arithmetic, Assignment, Unary Minus, Relational, Logical, Boolean, Bitwise, Membership, Identity, Operator precedence and Associativity

### Labs / Practicals:

1. Write and execute a C program for Transpose of a square matrix
2. Write and execute a C program to demonstrate call by value.
3. Write and execute a C program to declare, and initialize an UNION.
4. Write and execute a C program to demonstrate parameterized Macro substitution
5. Write a program to add the contents at the end of the file.
6. Write simple Python program to calculate equivalent registers connected in series and parallel.
7. Write python programs to define function with arguments. a) Calculate factorial of a number b) Swapping of two

variables.

### References:

1. Yashwant Kanetkar,Let Us Python, BPB, ISBN: 978-9391392253
2. Martin C. Brown, Python: The Complete Reference, Tata McGraw Hill ISBN: 9789387572942
3. Y. Kanetkar, Let Us C: Authentic guide to C programming language.
5. E. Balagurusamy, Object Oriented Programming With C++

| Course Code | Course Name | L | T | P | Cr |
|---|---|---|---|---|---|
| DASH250016 | Communication Skills | 1 | 0 | 2 | 2 |

### Course Outcomes:

CO1 : Understand the need and importance of English language.
CO2 : Develop proficiency in the language.
CO3 : Able to use technology to enrich communication skills.
CO4 : Gain self-confidence with improved command over English.
CO5 : Apply English communication skills effectively in academic, professional, and social environments.

### Module 1:

Communication Fundamentals
Communication ;Meaning, Process and importance of Communication, Types and Channels of communications , Scope and Significance of Listening ,Speaking, Reading, Writing Skills.

### Module 2:

Writing Skills
Basics of Grammar: Parts of Speech, Uses of Tenses, Active Passive, Narration.

### Module 3:

Vocabulary Building
Word Formation, Synonyms , Antonyms, Words often Confused, One Word Substitutes, Idioms and Phrasal verbs, Abbreviations of Scientific and Technical Words.

### Module 4:

Speaking Skills
Introduction to Phonetic Sounds & Articulation, Word Accent, Rhythm and Intonation, Interpersonal Communication, Oral Presentation, Body Language and Voice Modulation (Para linguistics and Non Verbal), Negotiation and Persuasion, Group Discussion, Interview Techniques (Telephonic and Video conferencing).

### Module 5:

Technical Writing
Job application, CV writing, Business Letters, Notices, Report Writing, Minutes, E-mail Etiquette, Blog Writing.

### Labs / Practicals:

1. Introducing Oneself, Exercise on Parts of Speech & Exercise on Tense.
2. Exercise on Agreement, Narration, Active Passive Voice & Dialogue Conversation.
3. Exercise on Writing Skills and Listening Comprehension.
4. Practice of Phonemes, Word Accent, Intonation, JAM Session.
5. Individual Presentation, Extempore and Picture Interpretation.
6. Vocabulary Building Exercises (One Word Substitute, Synonyms, Antonyms, Words Often Confused etc.) & Group Discussion.

### References:

1. "The Essence of Effective Communication", Ludlow R. and Panton F., Pubs: Prentice Hall.
2. “Effective Communication Skills”, Kulbhushan Kumar, Khanna Publishing House.
3. “A University Grammar of English”, Quirk R. and Sidney G., Pubs: Pearson Education.
4. “High School English Grammar”, Wren and Martin, Pubs: S. Chand & Company Ltd.
5. “Essentials of Business Communication”, Guffrey M.E., Pubs: South-Western College Publishing.
6. “Technical Communication: Principles and Practice”, Raman M. and Sharma S., Pubs: Oxford University Press.
7. “Effective Business Communication”, Rodrigues M.V., Pubs: Concept Publishing Company, Delhi.

8. “English Vocabulary in Use”, McCarthy M. and Felicity O’ Dell.

| Course Code | Course Name | L | T | P | Cr |
|---|---|---|---|---|---|
| DASH250101 | Sports and Yoga | 0 | 0 | 4 | 2 |

### Course Outcomes:

CO1: Understand the principles of physical education, fitness, wellness, and positive lifestyle habits essential for maintaining overall health.

CO2: Demonstrate knowledge of basic anatomy, physiology, and posture correction, and apply this understanding to enhance body mechanics and prevent injuries.

CO3: Practice fundamental yoga postures, breathing techniques, and relaxation methods to improve flexibility, concentration, and mental well-being.

CO4: Apply yoga and fitness practices to manage lifestyle-related health conditions and promote physical and psychological balance.

CO5: Participate actively in sports and fitness activities, demonstrating teamwork, discipline, and awareness of safety, nutrition, and lifelong fitness practices.

### Module 1:

Foundations of Physical Education and Fitness
Meaning & definition of Physical Education, Aims & Objectives of Physical Education, Changing trends in Physical Education, Meaning & importance of Physical Fitness and Wellness, Components of physical fitness and health-related fitness, Components of wellness, Concept of positive lifestyle, Preventing health threats through lifestyle changes.

### Module 2:

Anatomy, Physiology and Posture
Introduction to Anatomy and Physiology, Importance of Anatomy and Physiology in Physical Education, Effects of exercise on circulatory, respiratory, and neuromuscular systems, Meaning & concept of posture, Causes and correction of bad postures, Postural deformities: Knock knee, Flat foot, Lordosis, Kyphosis, etc., Weight training: pros and cons.

### Module 3:

Yoga-Theory and Practice
Meaning & importance of Yoga, Elements of Yoga: Asanas, Pranayama, Meditation, Kriyas, Yoga for concentration and related Asanas (Sukhasana, Tadasana, Padmasana, Shashankasana), Relaxation techniques (Yog-Nidra).

### Module 4:

Yoga for Lifestyle and Health Conditions
Preventive role of Asanas, Yoga for hypertension, obesity, back pain, diabetes, asthma, Benefits, procedures & contraindications for relevant Asanas (e.g., Bhujangasana, Pavan Muktasana, Tadasana, Chakrasana, etc.).

### Module 5:

Games and Sports Fundamentals
Select any one sport: Athletics, Basketball, Kabaddi, etc., History and development of the sport, General rules and regulations, Specifications of playfields and equipment, Major tournaments, venues, and personalities, Importance of proper sports gear.

### Labs / Practicals:

1. Practice of Sukhasana, Padmasana, and Tadasana for improving concentration and posture.

2. Shashankasana and Bhujangasana for relaxation and spinal flexibility.
3. Introduction to and practice of basic Pranayama techniques (Anulom-Vilom, Bhramari).
4. Practice of Yog Nidra (deep relaxation technique).
5. Asanas for lifestyle diseases: practice of Pavan Muktasana, Chakrasana, and Tadasana for obesity and diabetes control.
6. Demonstration and correction of bad postures (knock knee, flat foot, lordosis, kyphosis).
7. Light warm-up and stretching routines before yoga or sports activities.
8. Basic fitness tests: measurement of flexibility, endurance, strength, and body composition.
9. Participation in any one selected sport: demonstration of rules, gear, warm-up drills.
10. Practice of the chosen sport (e.g., Athletics, Basketball, Kabaddi) including basic skills and gameplay
11. Group activity: Demonstration of importance of proper sports gear and warm-up before sports.
12. Poster or presentation on a Yoga-based lifestyle for common health issues.
13. Introduction to circuit training and its benefits for physical fitness.
14. Meditation sessions and reflective journaling on mental well-being.
15. Nutritional awareness activity: discussion on diet for fitness and yoga lifestyle.

## References:

1. Modern Trends and Physical Education by Prof. Ajmer Singh.
2. Light On Yoga by B.K.S. Iyengar.
3. Health and Physical Education – NCERT (11TH and 12th Classes).
4. Back to health through YOGA by Ramesh Bijlani.

| Course Code | Course Name | L | T | P | Cr |
|---|---|---|---|---|---|
| DCSE250030 | Fundamentals of Computer Organization and Architecture | 2 | 0 | 4 | 4 |

### Course Outcomes:

CO1: Apply number systems, Boolean algebra, and logic minimization techniques to design and analyze combinational and sequential circuits.
CO2: Explain computer organization concepts including functional units, Von Neumann architecture, data representation, and micro-operations.
CO3: Analyze control unit design using microprogramming and apply arithmetic algorithms with pipelining and parallel processing techniques.
CO4: Develop assembly language programs for a chosen microprocessor using arithmetic, logical, branching, and procedural constructs.
CO5: Demonstrate memory organization and perform interfacing of RAM, ROM, I/O devices, and peripherals with processors.

### Module 1:

Data and number systems, Binary representation, codes and their conversions: BCD, Octal, Hexadecimal, Gray, Signed binary number representation with 1's and 2's complement methods. Binary arithmetic Boolean algebra, logic gates and circuits, Minimization of logic expressions by algebraic method, K-map method.
Combinational circuits: adder, subtractor, encoder, decoder, comparator, multiplexer, de-multiplexer, parity generator, Design of combinational circuits-Programming logic devices. Flip Flops, various types of Registers and counters and their design, sequential circuits.

### Module 2:

Structure of Computers: Computer Functional units, Von-Neumann architecture, Bus structures, Basic Operational Concepts, Data representation (Fixed and Floating point), Error detecting codes. Register Transfer and Micro Operations: Register transfer, Bus and memory transfers, Arithmetic micro-operations, Logic micro-operations, Shift micro-operations and Arithmetic logic shift unit.

### Module 3:

Micro Programmed Control: Control memory, Address sequencing and design of control unit. Computer Arithmetic: Addition and Subtraction, Multiplication and Division algorithms, Floating-point arithmetic operation, Arithmetic Pipeline, Instruction Pipeline, RISC Pipeline Vector Processing, Array Processors.

### Module 4:

Introduction to Microproccessor Architecture:
Instruction Set Architecture design principles from programmer's perspective. One example microprocessor (Intel, ARM, etc). Assembly Language Programming: Simple programs, Assembly language programs involving logical, branch and call instructions, sorting, evaluation of arithmetic expressions, string manipulation, assembler directives, procedures and macros.

### Module 5:

Memory and Digital Interfacing: addressing and address decoding, interfacing RAM, ROM, EPROM, programmable peripheral interface, various modes of operation and interfacing to processor, inter-facing keyboard, displays, etc.

### Labs / Practicals:

1. Implement Logic gates using NAND and NOR gates.
2. Design a Full adder using gates
3. Design and implement the 4:1 MUX, 8:1 MUX using gates /ICs.
4. Design and Implement a 3 to 8 decoder using gates
5. Implement a C program to convert a Hexadecimal, octal, and binary number to decimal number vice versa

6. Implement a C program to perform Binary Addition & Subtraction
7. Implement a C program to perform Multiplication of two binary numbers.
8. Implement a C program to perform Multiplication of two binary numbers (signed) using Booth's Algorithms.
9. Implement a C program to perform division of two binary numbers (Unsigned) using restoring division algorithm.
10. Implement a C program to perform division of two binary numbers (Unsigned) using non-restoring division algorithm.

## References:

1. Digital principles & Applications : Albert Paul Malvino & Donald P. Leach
2. Fundamentals of Logic Design : H. Roth Jr. Jaico
3. Computer System Architecture : M. Moris Mano, Pearson/PHI, India.
4. Microprocessors Interface : Douglas V. Hall, Tata McGraw-Hill.
5. Computer Organization : Carl Hamacher, Zvonks Vranesic, SafeaZaky, McGraw-Hill
6. Advanced Microprocessors and Peripherals-Architecture, Programming and interfacing, by:- A. K. Ray, K. M. Bhurchandi, Tata McGraw-Hill, New Delhi, India.

| Course Code | Course Name | L | T | P | Cr |
|---|---|---|---|---|---|
| DCSA250057 | Object Oriented Programming using C++ | 3 | 0 | 2 | 4 |

### Course Outcomes:

CO1. Describe the different features of Object Oriented Programming.
CO2. Write programs to Implement the concepts of classes and objects.
CO3. Create new classes using the concepts of inheritance.
CO4. Apply knowledge of Polymorphism to solve real life problems.
CO5. Implement exception handling mechanism.

### Module 1:

Introduction to Object Oriented Programming: Object Oriented Paradigm Objects and Classes, Features Object oriented Programming, Structured Vs Object Oriented Development, Features of Object Oriented Languages, Applications ofObjectOriented Programming, Merits and Limitations of Object Oriented Programming.Basic Data types, Basic Type modifiers, Derived Data types, Variables, Storage class specifiers, Initializing variables, Operators, Unformatted Console and stream I/O Functions, Formatted Console I/O Functions

### Module 2:

Classes and Objects: Classes ,Class Members and Creating Objects, Member functions, Member Access Specifiers (public, private, protected), Static class member, Inline Functions, Arrays within a Class and Array of Objects, Passing Objects as function arguments and returning object from a function,Constructors, Overloaded Constructors, Null Contradictor, Copy Constructor, Destructors Constraints on Constructors and Destructors

### Module 3:

Inheritance: Base and Derived classes, Accessing Base class members and Access Control, Overriding member functions, Multi Level, Multiple, Hierarchical& Hybrid Inheritance, Virtual Base Class.

### Module 4:

Polymorphism: Fundamental of Polymorphism, Overloading Functions, Overloading Operators (Unary, binary, string manipulation using operator), Pointer to object and derived class, 'This' pointer, Virtual Functions, Early and Late Binding, Rules of Virtual Functions, Pure Virtual Function, Friend Functions.

### Module 5:

File Handling, Exception Handling & Templates: Basic File Operations, File Handling, Classes for file stream operation, Opening and Closing Files, File modes File, Introduction to Exception Handling, Catching Class Types, Multiple Catch Handlers, Exception Specification, Generic Functions/Function Templates, Template Arguments.

### Labs / Practicals:

1. Write a C++ program to demonstrate class and object creation.
2. Implement constructors and destructors in a C++ class.
3. Demonstrate function overloading and operator overloading in C++.
4. Create a program to implement single and multilevel inheritance.
5. Use virtual functions to implement runtime polymorphism.
6. Design a template class for stack or queue operations.
7. Demonstrate file handling using file streams in C++.
8. Implement a program using exception handling with multiple catch blocks.

### References:

1. Object Oriented Programming With C++, E Balagurusamy
2. Object oriented programming in C++ , Robert Laffore
3. Introduction To Programming With C++,  Diane Zak
4. Object oriented programming with C++ Reema Thareja

| Course Code | Course Name | L | T | P | Cr |
|---|---|---|---|---|---|
| DCSE250056 | Data Structures and Algorithms | 3 | 0 | 2 | 4 |

### Course Outcomes:

1)Classify Data structures
2)Describe Linear Data Structures
3)Explain Non-Linear Data Structures
4)Explain basic algorithmic concepts and recursion
5)Apply different Sorting and Searching Algorithms

### Module 1:

Introduction to Data Structures:Basic Terminology, Classification of Data Structures, Operations on Data Structures

### Module 2:

Linear Data Structures: Arrays: Introduction to Arrays, Representation in Memory, Operations on an Array, Two Dimensional Arrays
Linked Lists: Singly Linked List, Representation in Memory, Operations on a Single Linked List, Circular Linked Lists, Doubly Linked Lists.
Stacks: Introduction to Stacks, Array Representation of Stacks, Operations on a Stack, Linked List Representation and Operations of Stack, Applications of Stacks-Infix-to-Postfix Transformation, evaluating Postfix Expressions.
Queues: Introduction to Queues, Array Representation of Queues, Operations on a Queue,
Linked List Representation and Operations of Queue, Types of Queues-DeQueue, Circular Queue, Applications of Queues-Round Robin Algorithm

### Module 3:

Non Linear Data Structures:
Trees: Basic Terminologies, Definition and Concepts of Binary Trees, Representations of a Binary Tree using Arrays and Linked Lists, Operations on a Binary Tree-Insertion, Deletion, Traversals, Types of Binary Trees.
Graphs: Graph Terminologies, Representation of Graphs- Set, List, Matrix, Graph Traversals

### Module 4:

Introduction to Algorithms:
Algorithms and flow charts, Time & Space complexity (definition only)
Recursion: Basic concepts and examples of recursion e.g. factorial problem, Fibonacci sequence.

### Module 5:

Sorting & Searching Algorithms: Sorting Algorithms: Algorithms and their analysis (time and space) — Bubble sort, Selection Sort, Insertion Sort, Quick Sort, Merge Sort, Heap sort and Radix Sort
Searching Algorithms: Linear search — Binary search –Concept of Hashing.

### Labs / Practicals:

1.Write a program to implement array
2.Write a program to add two matrices using two dimensional arrays
3.Write a program using recursive and non-recursive functions to perform search operation in a given list of integers using linear search technique
4.Write a program to implement search operation in a given list of integers using binary search technique

5.Write a program to implement insertion sorting for a given random data
6.Write a program to implement bubble sorting for a given random data
7.Write a program to implement quick sorting for a given random data
8.Write a program to implement selection sorting for a given random data.

## References:

1)Data Structures, R.S. Salaria, Khanna Book Publishing, New Delhi
2)Data Structures Using C, Reema Thareja, Oxford University Press India.
3)Classic Data Structures, SamantaDebasis, Prentice Hall of India.
4)Fundamentals of Data Structure in C, Horowitz, Ellis, Sahni, Sartaj, AndersonFreed, Susan, University Press, India.
5)Data Structures: A Pseudo code approach with C, Richard F. Gilberg, Behrouz A. Forouzan, CENGAGE Learning, India.
6)Data Structures and Algorithms: Concepts, Techniques and Applications, G. A. V. Pai, McGraw- Hill Education, India.
7)Introduction to Algorithms, T.H. Cormen, C.E. Leiserson, R. L. Rivest, C. Stein, MIT Press

| Course Code | Course Name | L | T | P | Cr |
|---|---|---|---|---|---|
| DCSE250102 | Operating Systems | 3 | 0 | 2 | 4 |

### Course Outcomes:

CO1: Understand the basics of operating systems like kernel, shell, types and services of operating systems.
CO2:Understand the concept of program, process and thread and analyse various CPU scheduling Algorithms.
CO3: Describe and analyse the memory management and its allocation policies.
CO4: Understand the issues related to file system interface and implementation.
CO5: Understand disk management and explain disk scheduling algorithms for better utilization of external memory.
6: Configure OS in an efficient and secure manner.

### Module 1:

Introduction: Overview of Operating System, basic concepts, UNIX/LINUX Architecture, Kernel, services and systems calls, system programs.

### Module 2:

Process Management and Memory management: Process Management: Process concepts, operations on processes, IPC, Process Scheduling, Multithreaded programming. Memory management: Memory allocation, Swapping, Paging, Segmentation, Virtual Memory, various faults.

### Module 3:

File management: Concept of a file, access methods, directory structure, file system mounting, file sharing and protection, file system structure and implementation, directory implementation, free space management, efficiency and performance. Different types of file systems.

### Module 4:

I/O System: Mass storage structure - overview, disk structure, disk attachment, disk scheduling algorithms, swap space management, RAID types.

### Module 5:

OS Security: Authentication, Access Control, Access Rights, System Logs

### Labs / Practicals:

1. Revision practice of various commands like man, cp, mv, ln, rm, unlink, mkdir, rmdir, etc and many more that were learnt in IT Workshop course and later.
2. Implement two way process communication using pipes.
3. Implement message queue form of IPC
4. Implement shared memory and semaphore form of IPC
5. Simulate the CPU scheduling algorithms - Round Robin, SJF, FCFS, priority
6. Simulate all FIFO Page Replacement Algorithm using C program
7. Simulate all LRU Page Replacement Algorithms using C program
8. Simulate Paging Technique of Memory Management
9. Practice various commands/utilitiessuch as catnl, uniq, tee, pg, comm, cmp, diff, tr, tar, cpio, mount, umount, find, umask, ulimit, sort, grep, egrep,fgrep cut, paste, join, du, df , ps, who, etc and many more.

### References:

1. "Operating System Concepts (Tenth Edition)" by Abraham Silberschatz, Peter Baer Galvin, and Greg Gagne
https://os-book.com/OS10/index.html

2. "Operating Systems: Internals and Design Principles (Ninth Edition)" by William Stallings
http://williamstallings.com/OperatingSystems

3. "Modern Operating Systems (Fifth Edition)" by Andrew Stuart Tanenbaum and Herbert Bos
https://dl.acm.org/doi/10.5555/2655363

4. "Operating System Design: The Xinu Approach (Second Edition)" by Douglas Earl Comer
https://www.oreilly.com/library/view/operating-system-design/9781498712439

5. "The Design of the UNIX Operating System" by Maurice J. Bach
https://dl.acm.org/doi/10.5555/8570

6. "Advanced Programming in the UNIX Environment (Third Edition)" by William Richard Stevens and Stephen A. Rago
http://www.apuebook.com/apue3e.html

7. "Beej's Guide to Interprocess Communication" by Brian Hall
https://beej.us/guide/bgipc/pdf/bgipc_a4_c_2.pdf

| Course Code | Course Name | L | T | P | Cr |
|---|---|---|---|---|---|
| DCSE250060 | Database Management Systems | 2 | 0 | 4 | 4 |

### Course Outcomes:

CO1. Understand the basic database concepts, data models, schemas and instances.
CO2. Learn database design using Entity Relationship model and learn the use of constraints and relational algebra operations.
CO3. Gain proficiency in SQL and construct queries using SQL.
CO4. Understand the importance of normalization in database design.
CO5. Understand transaction processing, concurrency control and recovery

### Module 1:

Introduction to Databases, File systems vs. Database systems, Characteristics of DBMS, Applications of DBMS, Advantages of using a DBMS, Database users and administrators, Three-level architecture of DBMS (ANSI/SPARC architecture), Data independence, Database schema, Instance, Generalization, Specialization, Structure of a DBMS, Overview of DBMS software (e.g., MySQL, SQL Server, Oracle, PostgreSQL)

### Module 2:

Introduction to Data models: Hierarchical, Network, Relational, Object-oriented, Data Modeling using Entity Relationship (ER) model: entities, entity types, attributes, relationships, relationship types, E/R diagram notation, examples.
Relational data model: Concepts, schema, instances, keys, constraints, Relational algebra: Select, project, union, set difference, Cartesian product, joins, ER to Relational mapping, Relational algebra and relational calculus, Tuple Relational Calculus, Domain Relational Calculus

### Module 3:

Structured Query Language (SQL): Basics of SQL, DDL, DML,DCL, structure – creation, alteration, defining constraints – Primary key, foreign key, unique, not null, check, IN operator, Functions - aggregate functions, Built-in functions – numeric, date, string functions, set operations, sub-queries, correlated sub-queries, Use of group by, having, order by, join and its types, Exist, Any, All , view and its types. Transaction control commands – Commit, Rollback, Save point, Cursors, Indexes, stored procedures, Triggers

### Module 4:

Normalization – Introduction, Non loss decomposition and functional dependencies, First, Second, and third normal forms – dependency preservation, Boyce/Codd normal form.
Higher Normal Forms - Introduction, Multi-valued dependencies and Fourth normal form, Join dependencies and Fifth normal form.

### Module 5:

Transaction management and Concurrency control: Transaction processing and Error recovery - concepts of transaction processing, ACID properties, and serializability concurrency control, Lock based concurrency control (2PL, Deadlocks), Time stamping methods, optimistic methods, and database recovery management. Error recovery and logging, undo, redo, undo-redo logging and recovery methods, Introduction to database security concepts.

### Labs / Practicals:

1. Create a database, define tables using DDL, apply constraints (Primary, Foreign, Not Null, Unique).
2. Insert, update, delete, and retrieve records using SQL DML commands.
3. Write SQL queries using WHERE, ORDER BY, GROUP BY, HAVING, LIKE, etc.
4. Write SQL queries to demonstrate the use of INNER JOIN, LEFT OUTER JOIN, RIGHT OUTER JOIN, and SELF JOIN using appropriate sample tables.
5. Write nested subqueries with IN, EXISTS, and ANY/ALL.

6. Draw ER diagram for a sample application (e.g., Library or Hospital), map it to relational schema.
7. Normalize sample unstructured data up to 3NF or BCNF. Implement before-and-after schemas in SQL.
8. Create and execute stored procedures/functions for common operations.

## References:

1. Silberschatz, Korth, Sudarshan – Database System Concepts, McGraw Hill Year: 2020 (7th edition).
2. Elmasri and Navathe – Fundamentals of Database Systems, Pearson Edition 2016
3. Raghu Ramakrishnan, Johannes Gehrke – Database Management Systems, McGraw Hill
4. Database Systems Concepts Author H.f.Korth and Silberschatz Publisher McGraw Hill
5. Database System Author Hector Garcia-Molina, Jeff Ullman, and Jennifer Widom Publisher Pearson Edition 2nd Edition
6. C.J. Date – An Introduction to Database Systems, Pearson

| Course Code | Course Name | L | T | P | Cr |
|---|---|---|---|---|---|
| DCSE250035 | Computer Networks | 3 | 0 | 2 | 4 |

### Course Outcomes:

CO1: Understand Network Basics – Learn network models, topologies, and transmission media.
CO2: Analyse Data Link Layer – Evaluate error detection, MAC protocols, and wireless communication.
CO3: Implement Network Layer Functions – Apply IP addressing, subnetting, and routing algorithms.
CO4: Apply Transport & Application Layer Concepts – Compare TCP/UDP and analyse key protocols.
CO5: Demonstrate Security & Emerging Technologies – Identify threats, implement cryptography, explore SDN, IoT, and cloud networking.

### Module 1:

Introduction to Computer Networks Overview of Computer Networks, Network Types: LAN, WAN, MAN, PAN, Network Topologies: Bus, Star, Ring, Mesh, Hybrid, OSI Model & TCP/IP Model – Layers and Functions, Transmission Media: Wired & Wireless Technologies, Switching Techniques: Circuit, Packet, and Message Switching

### Module 2:

Data Link Layer & MAC Protocols
Error Detection and Correction: Parity, Checksum, CRC, Hamming Code, Flow Control Mechanisms: Stop-and-Wait, Sliding Window Protocol, MAC Protocols: ALOHA, CSMA/CD, CSMA/CA, Ethernet Standards (IEEE 802.3), VLANs, Spanning Tree Protocol, Wireless LAN (Wi-Fi), Bluetooth, and Mobile Networks

### Module 3:

Network Layer & Routing Algorithms
IPv4 and IPv6 Addressing, Subnetting, and CIDR, Routing Algorithms: Distance Vector, Link-State, and Hybrid Routing, Routing Protocols: RIP, OSPF, EIGRP, and BGP, Congestion Control Mechanisms and Quality of Service (QoS), Network Address Translation (NAT) and DHCP

### Module 4:

Transport Layer & Application Layer Protocols
TCP and UDP: Connection Establishment, Flow Control, and Error Control, Congestion Control: TCP Reno, TCP Tahoe, and Fair Queuing, Application Layer Protocols: HTTP, HTTPS, FTP, SMTP, POP3, IMAP, DNS and URL Resolution, Peer-to-Peer Networks, Security in Transport & Application Layers: TLS, SSL, Firewalls

### Module 5:

Network Security and Emerging Trends
Cryptography Fundamentals: Symmetric and Asymmetric Encryption, Authentication and Digital Signatures, Network Security Attacks: DoS, DDoS, Phishing, Man-in-the-Middle, Wireless and IoT Security Challenges, Software-Defined Networking (SDN) and Cloud Networking, Introduction to 5G, Edge Computing, and Quantum Networking

### Labs / Practicals:

Preparing

### References:

1. "Computer Networking: A Top Down Approach (Eighth Edition)" by James F. Kurose and Keith W. Ross

https://gaia.cs.umass.edu/kurose_ross/about.php

2. "An Introduction to Computer Networks (Second Edition)" by Peter Lars Dordal
https://intronetworks.cs.luc.edu/current2/ComputerNetworks.pdf

3. "Computer Networking: Principles, Protocols and Practice (Third Edition)" by Olivier Bonaventure
https://github.com/cnp3/ebook/releases/download/draft-3rd/CNP3-2021.pdf

4. "Computer Networks: A Systems Approach (Sixth Edition)" by Larry L. Peterson and Bruce S. Davie
https://github.com/SystemsApproach/book/releases/download/v6.1/book.pdf

5. "Computer Networks (Fifth Edition)" by Andrew Stuart Tanenbaum and David J. Wetherall
https://www.oreilly.com/library/view/computer-networks-fifth/9780133485936

6. "Internetworking with TCP/IP: Principles, Protocols, and Architecture (Sixth Edition)" by Douglas Earl Comer
https://www.oreilly.com/library/view/internetworking-with-tcpip/9780137464197

7. "UNIX Network Programming: Volume 1 (Third Edition)" by William Richard Stevens, Bill Fenner, and Andrew M. Rudoff
https://www.oreilly.com/library/view/the-sockets-networking/0131411551

8. "Computer and Network Organization: An Introduction" by Maarten van Steen and Henk Sips
https://www.distributed-systems.net/index.php/books/computer-and-network-organization

9. "Beej's Guide to Network Programming" by Brian Hall
https://beej.us/guide/bgnet/pdf/bgnet_a4_c_2.pdf

| Course Code | Course Name | L | T | P | Cr |
|---|---|---|---|---|---|
| DASH250091 | Professional Values and Ethics | 2 | 0 | 0 | 2 |

## Course Outcomes:

CO1: Describe profession, professionalism, its challenges, effective communication, and the role of regulatory bodies and professional organizations related to engineering.
CO2: Analyze professional values such as integrity, responsibility, respect, fairness, and accountability, and demonstrate their application in professional practice.
CO3: Apply knowledge of ethics and bioethics in ethical decision-making in collaboration with professional teams.
CO4: Evaluate responsibilities related to safety, risk management, and intellectual property in professional practice.
CO5: Examine global ethical issues and professional responsibilities in technological and research contexts.

## Module 1:

Values
Definition and characteristics of values, value clarification, personal and professional values, professional socialization: integration of professional and personal values, importance of professional values in engineering and technology, core values – integrity, honesty, trust, respect, responsibility, fairness, accountability, ethical conduct in workplace relationships and decision-making.

## Module 2:

Engineering Ethics
Senses of engineering ethics, variety of moral issues, types of inquiry, moral dilemmas, moral autonomy, Kohlberg's theory, Gilligan's theory, consensus and controversy, models of professional roles, theories about right action – self-interest, customs and religion, application of ethical theories, valuing time, cooperation, commitment, case studies.

## Module 3:

Engineering as Social Experimentation
Engineering as social experimentation, framing the problem, determining the facts, codes of ethics, clarifying concepts, application issues, common ground, general principles – utilitarian thinking and respect for persons, case studies.

## Module 4:

Engineers Responsibility for Safety and Risk
Safety and risk, assessment of safety and risk, risk-benefit analysis, reducing risk, safety and the engineer – designing for safety, Intellectual Property Rights (IPR).

## Module 5:

Global issues
Globalization, cross-cultural issues, environmental ethics, computer ethics, computers as instruments of unethical behavior, computers as objects of unethical acts, autonomous systems, computer codes of ethics, weapons development, ethics and research, analyzing ethical problems in research, case studies.

## Labs / Practicals:

N/A

## References:

1. M. Govindarajan, S. Natarajan, V.S. Senthil Kumar – Engineering Ethics Includes Human Values, PHI Learning Pvt. Ltd.
2. Charles E. Harris, Michael S. Pritchard, Michael J. Rabins – Engineering Ethics, Cengage Learning.
3. Mike W. Martin, Roland Schinzinger – Ethics in Engineering, Tata McGraw Hill.

4. A.R. Aryasri, Dharanikota Suyodhana – Professional Ethics and Morals, Maruthi Publications.
5. A. Alavudeen, R. Kalil Rahman, M. Jayakumaran – Professional Ethics and Human Values, Laxmi Publications.
6. D.R. Kiran – Professional Ethics and Human Values.
7. P.S.R. Murthy – Indian Culture, Values and Professional Ethics, BS Publications

| Course Code | Course Name | L | T | P | Cr |
|---|---|---|---|---|---|
| DASH250052 | Environmental Science | 2 | 0 | 0 | 2 |

### Course Outcomes:

CO1: Understand the importance and dimension of a healthy environment, become environmentally conscious, skilled and responsible in all their actions with a concern for sustainable development.
CO2: Comprehend the significance and issues related to ecosystems, natural resources and bio-diversity and become aware of the need and ways to protect/ preserve them.
CO3: Grasp the issues related to environmental pollution, solid waste management and climate change, and become conscious and proactive in the discharge of their responsibilities towards the environment.
CO4: Become aware and appreciate the values and concerns of environmental movements and policies and the role of communities, and act responsibly on environment-related issues.
CO5: Understand environmental policies, ethics, and community roles in sustainable development.

### Module 1:

Introduction to Environmental Studies and Ecosystems:
Environmental Studies: Importance and scope, Multidisciplinary nature, Concept of sustainability and sustainable development, Ecosystems: Concept, Structure and Function, Pond and forest ecosystems, Food chains and food webs, Concept of ecological succession, Bio-geographical zones of India, Levels of biological diversity: Genetic, Species, and Ecosystem.

### Module 2:

Biodiversity and Conservation:
Biodiversity Hotspots with special reference to India, Threats to biodiversity, Conservation of biodiversity: In-situ and Ex-situ, Endangered and endemic species: Concept Afforestation: Social forestry, Agroforestry, Green belt.

### Module 3:

Pollution and Waste Management:
Air, water, and noise pollution: Causes, effects, and control measures, Climate change, global warming, ozone depletion, acid rain: Impact on humans and agriculture, Solid waste management: biodegradable and non-biodegradable waste.
- Segregation of domestic waste at source
- Impact of plastic on human and animal health

### Module 4:

Natural Resources and Disaster Management:
Land resources and land-use changes, Land degradation, soil erosion, and desertification, Water: Use and over-exploitation, Water conservation : Rainwater harvesting, Watershed management : Meaning and importance, Energy resources: Renewable and non-renewable, Alternate energy sources, Disaster management – Types and self-protection during disasters Environmental Policy, 2006 : Provisions and importance; Environmental Impact Assessment – Concept; Swachh Bharat Mission Objectives; International agreements , Montreal and Kyoto protocols.

### Module 5:

Environmental Policies and Human Role
Population growth: Impact on environment, health, and welfare, Environmental ethics : Religion and cultural role, Environmental movements: Chipko, Narmada Bachao, Silent Valley, Bishnoi's, Community initiatives:
Salu Marada Thimmakka, Sacred Groves (Devarakadu), Environment Protection Act, Biodiversity Act (2002), National Environmental Policy (2006), Environmental Impact Assessment, Swachh Bharat Mission, International agreements: Montreal and Kyoto Protocols.

### Labs / Practicals:

NA

## References:

1. Asthana, D. K. .Text Book of Environmental Studies. S. Chand Publishing.
2. Basu, M., Xavier, S. . Fundamentals of Environmental Studies, Cambridge University Press, India
3. Basu, R. N. Environment. University of Calcutta, Kolkata
4. Bharucha, E. Textbook of Environmental Studies for Undergraduate Courses. Universities Press, India.
5. De, A.K. Environmental Chemistry, New Age International, .
6. Mahapatra, R., Jeevan, S.S., Das, S. (Eds) . Environment Reader for Universities, Centre for Science and Environment.
7. Masters, G. M., &Ela, W. P. .Introduction to environmental engineering and science. Englewood Cliffs, NJ: Prentice Hall.
8. Odum, E. P., Odum, H. T., & Andrews, J. Fundamentals of ecology. Philadelphia: Saunders.
9. Sharma, P. D., & Sharma, P. D. .Ecology and environment. Rastogi Publications.

| Course Code | Course Name | L | T | P | Cr |
|---|---|---|---|---|---|
| DCSE250079 | Fundamentals of Programming in Python | 2 | 0 | 4 | 4 |

### Course Outcomes:

CO1: Develop algorithmic solutions to solve simple and well-defined computational problems effectively.
CO2: Write and demonstrate Python programs using basic statements, expressions, and syntax constructs.
CO3: Explain and implement control flow and function concepts in Python to solve programming tasks.
CO4: Apply Python data structures, including lists, tuples, and dictionaries, for managing compound data.
CO5: Describe and use file handling, exceptions, modules, and packages in Python-based problem-solving.

### Module 1:

Python Introduction, Technical Strength of Python, Introduction to Python Interpreter and program execution, Using Comments, Literals, Constants, Python's Built-in Data types, Numbers (Integers, Floats, Complex Numbers, Real, Sets), Strings (Slicing, Indexing, Concatenation, other operations on Strings), Accepting input from Console, printing statements, Simple 'Python' programs. Flow Chart Symbols, Basic algorithms/flowcharts for sequential processing, decision based processing and iterative processing.

### Module 2:

Assignment statement, expressions, Arithmetic, Relational, Logical, Bitwise operators and their precedence, Conditional statements: if, if-else, if-elif-else; simple programs, Notion of iterative computation and control flow –range function, While Statement, For loop, break statement, Continue Statement, Pass statement, else, assert. Lists, tuples, and dictionaries—slicing, indexing, concatenation, and mutability. Operations such as finding max, min, mean, linear search, and frequency count using dictionaries.

### Module 3:

Top-down approach of problem solving, Modular programming and functions, Function parameters, Local variables, the Return statement, DocStrings, global statement, Default argument values, keyword arguments, VarArgs parameters. Library function-input(), eval(),print(), Recursion.

### Module 4:

String Functions: count(), find(), rfind(), capitalize(), title(), lower(), upper(), swapcase(), islower(), isupper(), istitle(), replace(), strip(), lstrip(), rstrip(), aplit(), partition(), join(), isspace(), isalpha(), isdigit(), isalnum(), startswith(), endswith(), encode(), decode(), String: Slicing, Membership, Pattern Matching, Numeric Functions: eval(), max(), min(), pow(), round(), int(), random(), ceil(), floor(), sqrt(), Date & Time Functions.

### Module 5:

File opening in various modes and closing of a file, Reading from a file, Writing onto a file, File functions-open(), close(), read(), readline(), readlines(),write(), writelines(),tell(),seek(), Command Line arguments.

### Labs / Practicals:

1. Write a program to take name and age from user and display it.
2. Write a program illustrating the various operators in python (arithmetic, relational, logical, and bitwise operators).
3. Write a program to check the data type of given variables.
4. Write a program to swap two variables.
5. Write a program to check whether a number is even or odd.
6. Write a program to find the largest among three numbers.
7. Write a program to check if a number is positive, negative or zero.
8. Write a program to check whether a year is a leap year or not.
9. Write a program to calculate grade based on marks.
10. Write a program to print the multiplication table of a number.
11. Write a program to find the factorial of a number using a loop.

12. Write a program to display Fibonacci series up to n terms.
13. Write a program to find the sum of digits of a number.
14. Write a program to print all prime numbers in a given range
15. Write a Python function to produce pyramid triangles.
16. Write a program to count vowels and consonants in a string.
17. Write a program to reverse a string.
18. Write a program to check whether a string is a palindrome.
19. Write a program to find the length of a string without using len().
20. Write a program to remove all punctuation from a string.
21. Write a program to find the largest and smallest number in a list.
22. Write a program to add elements to a list and sort them.
23. Write a program to find the sum and average of list elements.
24. Write a program to find duplicates in a list.
25. Write a program to convert a list to a tuple and vice versa.
26. Write a function to calculate factorial of a number.
27. Write a function to check whether a number is prime.
28. Write a function that returns the square of a number.
29. Write a program with a function to count even and odd numbers in a list.
30. Write a recursive function to find the nth Fibonacci number.
31. Write a function that takes the lengths of three sides: side1, side2 and side3 of the triangle as the input from the user using input function and return the area of the triangle as the output. Also, assert that sum of the length of any two sides is greater than the third side.
32. Write a program to create and write to a text file.
33. Write a program to read contents of a file and display it.
34. Write a program to count the number of lines and words in a file.
35. Write a program to append data to an existing file.
36. Write a program to copy contents from one file to anothe

## References:

1. Python Network Programming Cookbook by Pradeeban Kathiravelu, Dr. M. O. Faruque Sarkar, PACKT.
2. Head First Python by Paul Berry, O'Reilly
3. Dive into Python by Mark Pilgrim, APress
4. Beginning Programming with Python Dummies by John Paul Meuller.

| Course Code | Course Name | L | T | P | Cr |
|---|---|---|---|---|---|
| DCSE250120 | Theory of Computation | 3 | 1 | 0 | 4 |

### Course Outcomes:

CO1. Explain the basic concepts of automata, formal languages, and grammar types.
CO2. Design finite automata (deterministic and non-deterministic) and regular expressions for recognizing regular languages.
CO3. Design regular expressions for describing simple patterns and languages.
CO4. Understand and design pushdown automata for basic context-free languages.

### Module 1:

Mathematical Preliminaries: Sets, Relations and Functions (Brief Discussion), Graphs, Trees. ,Strings and their properties: Definition, operation on strings, palindrome, prefix & suffix of a string, Levi theorem (Statement only), Terminal & Non-terminal symbol

### Module 2:

Definition of an Automaton: Definition of finite Automaton, Block diagram of finite Automaton, Transition system, Properties of Transition Functions, Acceptability of a string by Finite Automaton. Definition of DFA and NDFA, The equivalence of DFA and NDFA. Mealy and Moore machine.

### Module 3:

Formal Language: Concept of a language, Definition of a grammar, Language generated by a grammar (definition with application). Chomsky classification of languages (definition), Relation between the classified languages. Recursive and recursively enumerable set (definitions).

### Module 4:

Regular Sets & Regular Grammar: Definition of Regular expression and regular set, Identities of regular expressions
Relation between regular expression and finite automata, Transition system containing /\-moves (application), Conversion of Non-deterministic systems to deterministic system (application), Construction of finite automata equivalent to a regular expression (with application)

### Module 5:

Context-Free Languages & Pushdown Automata: Introduction – Definition – Derivation trees (Definitions & application) – Ambiguity in
CFG, Basic definition of PDA

### Labs / Practicals:

NA

### References:

1. "Foundations of Computation (Second Edition)" by Carol Critchlow and David Eck
https://math.hws.edu/FoundationsOfComputation/FoundationsOfComputation_2.3.2_8.5x11.pdf

2. "Automata Theory: An Algorithmic Approach" by Francisco Javier Esparza Estaun and Michael Blondin
https://michaelblondin.com/automata/files/book_authors.pdf

3. "Automata and Computability" by Dexter Campbell Kozen
https://www.cs.cornell.edu/kozen/Papers/481.pdf

4. "Introduction to Automata Theory, Languages, And Computation (First Edition)" by John Edward Hopcroft and Jeffrey David Ullman
http://infolab.stanford.edu/~ullman/ialc.html

5. "Introduction to the Theory of Computation (Third Edition)" by Michael Fredric Sipser
https://dl.acm.org/doi/10.5555/524279

6. "An Introduction to Formal Languages and Automata (Fifth Edition)" by Peter Linz
https://dl.acm.org/doi/10.5555/1995326

| Course Code | Course Name | L | T | P | Cr |
|---|---|---|---|---|---|
| DCSE250088 | Internet and Web Technologies | 3 | 0 | 2 | 4 |

### Course Outcomes:

CO 1: Explain Internet and WWW concepts, protocols, and tools required for building functional websites.
CO 2: Design and develop structured, interactive webpages using HTML elements, forms, tables, and frames.
CO 3: Apply JavaScript fundamentals, control structures, and objects to create dynamic and interactive web applications.
CO 4: Use CSS and DOM techniques to design styled, responsive, and dynamic web content effectively.
CO 5: Explain Internet security concepts, firewalls, and privacy issues in secure information management.

### Module 1:

The Internet and WWW: Understanding the WWW and the Internet, Emergence of Web, Web Servers, Web Browsers, Protocols, and Building Web Sites.

### Module 2:

HTML: Planning for designing Web pages, Model and structure for a Website, Developing Websites, Basic HTML using images links, Lists, Tables and Forms, Frames for designing a good interactive website.

### Module 3:

JAVA Script: Programming Fundamentals, Statements, Expressions, Operators, Popup Boxes, Control Statements, Try....Catch Statement, Throw Statement, Objects of
JavaScript: Date object, array object, Boolean object,
math object

### Module 4:

Cascading Style Sheets (CSS): External Style Sheets, Internal Style Sheets, Inline Style, The class selector, div & span tag.
DOM HTML DOM, inner HTML, Dynamic HTML (DHTML), DHTML form, XML DOM

### Module 5:

Internet Security Management Concepts: Overview of Internet Security, Firewalls, Internet Security, Management Concepts and Information Privacy and Copyright Issues

### Labs / Practicals:

1. Design a basic personal web page using HTML with text, images, and hyperlinks.
2. Create a registration form using HTML with input elements and submit button.
3. Design an HTML page using tables and nested lists for content organization.
4. Apply external and internal CSS to enhance the look and feel of a web page.
5. Use JavaScript to validate a form with fields like name, email, and password.
6. Create a dynamic web page that displays date and time using JavaScript Date object.
7. Demonstrate the use of DOM to modify HTML elements dynamically via JavaScript.
8. Build a simple webpage layout using div, span tags and CSS box model.

### References:

1. D. Comer, “The Internet Book”, 5th edition , 2019, Pearson Education.
2. Ivan Bayross, “HTML, DHTML, JavaScript, Perl CGI”, 4th (Revised) edition, 2022, BPB Publications.
3. Godbole AS & Kahate A, “Web Technologies”, 3rd edition, 2017, Tata McGraw Hill.
 Greenlaw R and Hepp E “Fundamentals of Internet and www” 2nd edition, 2007

| Course Code | Course Name | L | T | P | Cr |
|---|---|---|---|---|---|
| DCSA250071 | Software Engineering | 3 | 1 | 0 | 4 |

### Course Outcomes:

CO1: Understand Software Engineering Concepts – Explain core principles, ethical responsibilities, software processes, and critical system properties.
CO2: Analyze Software Requirements – Differentiate functional and non-functional requirements and apply requirements engineering processes.
CO3: Apply System Modeling & Project Management – Develop system models and manage software projects, including planning, scheduling, and risk management.
CO4: Design & Develop Software – Implement architectural and object-oriented design, agile methodologies, and software evolution strategies.
CO5: Perform Software Testing & Cost Estimation – Conduct verification, validation, software testing, team management, and cost estimation techniques

### Module 1:

Overview - Introduction, FAQs about software engineering, Professional and ethical responsibility; Socio-technical systems - Emergent system properties, Systems engineering; Critical systems and Software processes - Critical systems, A simple safety-critical system, System dependability, Availability and reliability; Software processes - Software process models, Process iteration, Process activities, The Rational Unified Process, Computer-Aided Software Engineering.

### Module 2:

Requirements - Software requirements, Functional and non-functional requirements, User requirements, System requirements, Interface specification, The software requirements document; Requirements engineering processes - Feasibility studies, Requirements elicitation and analysis, Requirements validation, Requirements management.

### Module 3:

System models and Project Management - System models, Context models, Behavioural models, Data models, Object models, Structured methods; Project management - Management activities, Project planning, Project scheduling, Risk management

### Module 4:

Software Design - Architectural design, Architectural design decisions, System organisation, Modular decomposition styles, Control styles; Object-oriented design - Objects and object classes, An object-oriented design process, Design evolution; Development - Rapid software development, Agile methods, Extreme programming, Rapid application development; Software evolution - Program evolution dynamics, Software maintenance, Evolution processes, Legacy system evolution.

### Module 5:

Verification and validation - Verification and validation, Planning verification and validation, Software inspections, Automated static analysis, Verification and formal methods; Software testing - System testing, Component testing, Test case design, Test automation; Management - Managing people, Selecting staff, Motivating people, Managing groups, The People Capability Maturity Model; Software cost estimation, Software productivity, Estimation techniques, Algorithmic cost modelling, Project duration and staffing

### Labs / Practicals:

N/A

### References:

1.Ian Sommerville: Software Engineering, 8th Edition, Pearson Education, 2007.
2.Roger S. Pressman: Software Engineering - A Practitioner’s Approach, 7th Edition, Tata McGraw Hill, 2007.
3.Pankaj Jalote: An Integrated Approach to Software Engineering, Wiley India, 2009.

| Course Code | Course Name | L | T | P | Cr |
|---|---|---|---|---|---|
| DOAI250013 | Machine Learning Fundamentals | 2 | 0 | 4 | 4 |

### Course Outcomes:

CO1. Understand and differentiate types of learning in machine learning.
CO2. Apply simple regression and classification techniques to practical problems.
CO3. Analyze and cluster data using unsupervised learning models.
CO4. Understand and implement Naive Bayes and basic sequence models.
CO5. Describe the structure of a neural network and use simple deep learning ideas.
CO6. Use basic ML libraries such as Scikit-learn for implementation.

### Module 1:

Introduction to Machine Learning:
Definition of ML, Applications and Use Cases, Types of Machine Learning (Supervised, Unsupervised, Reinforcement), Basic Learning Concepts, Concept and Hypothesis Space, Introduction to Learning Models

### Module 2:

Supervised Learning Methods:
Linear Regression, Overfitting and Underfitting, Cross Validation, Logistic Regression, K-Nearest Neighbors (KNN), Decision Trees, Random Forest, Model Evaluation Metrics (Accuracy, Precision, Recall)

### Module 3:

Unsupervised Learning Methods:
Clustering Algorithms: K-Means and Hierarchical Clustering, Dimensionality Reduction using PCA, Basics of Recommendation Systems

### Module 4:

Probability-Based Models:
Naïve Bayes Classifier, Maximum Likelihood, Basics of Bayesian Networks, Introduction to Hidden Markov Models (HMM) and Sequence Modeling

### Module 5:

Introduction to Neural Networks & Deep Learning:
Perceptron, Feedforward Networks, Activation Functions, Concept of Backpropagation, Limitations of Classical ML, Introduction to Deep Learning and Use Cases

### Labs / Practicals:

1. Introduction to Python libraries for ML (NumPy, Pandas, Matplotlib, Scikit-learn).
2. Build and test a Linear Regression model for house price prediction.
3. Apply Logistic Regression to binary classification (e.g., email spam detection).
4. Implement and compare K-Nearest Neighbors (KNN) with different k-values.
5. Train and evaluate a Decision Tree Classifier on a basic dataset.
6. Implement K-Means clustering on a synthetic dataset and visualize the clusters.
7. Apply Principal Component Analysis (PCA) on an image dataset for dimensionality reduction.
8. Use Naive Bayes Classifier for text classification (e.g., sentiment analysis).

9. Introduction to training a simple neural network using TensorFlow/Keras.

### References:

1. Aurelien Geron, Hands-On Machine Learning with Scikit-Learn and TensorFlow, O'Reilly, 2nd Ed.
2. Stephen Marsland, "Machine Learning: An Algorithmic Perspective", Chapman & Hall/CRC, 2nd Edition, 2014.

3. Kevin Murphy, Machine Learning: A Probabilistic Perspective, MIT Press.

Online Resources
1. https://scikit-learn.org
2. https://www.kaggle.com/learn/intro-to-machine-learning

| Course Code | Course Name | L | T | P | Cr |
|---|---|---|---|---|---|
| DCSA250003 | Mobile Application Development using Android | 2 | 1 | 2 | 4 |

### Course Outcomes:

CO1: Understand the architecture, lifecycle, and development environment of Android applications.
CO2: Design and build responsive user interfaces using Android layout and widget components.
CO3: Develop Android applications using activities, intents, and persistent data storage.
CO4: Integrate multimedia, location, and device services such as sensors and GPS in Android apps.
CO5: Deploy Android applications and apply debugging, testing, and publishing techniques.

### Module 1:

Introduction to Android and Development Environment
Overview of Mobile App Development Landscape
Introduction to Android OS: History, Features, and Architecture
Android Studio: Installation and Setup
Android Project Structure (Manifest, Java/Kotlin files, Res folder)
Activity Lifecycle
Hello World App
Emulator and Device Testing Basics

### Module 2:

User Interface Design and Layout Management
UI Components: TextView, EditText, Button, ImageView, CheckBox, RadioButton, Spinner
Layouts: LinearLayout, RelativeLayout, ConstraintLayout, FrameLayout
Event Handling and Listeners
Toasts, Dialog Boxes, and Snackbars
RecyclerView and Adapters
Menus: Options Menu, Context Menu

### Module 3:

Activities, Intents, and Data Persistence
Intents: Explicit and Implicit
Activity Communication (startActivityForResult)
Fragments: Introduction and Usage
Shared Preferences and Internal Storage
SQLite Database: CRUD Operations
Room Persistence Library (Basics)

### Module 4:

Advanced Features and Device Integration
Multimedia: Audio, Video Playback
Camera Integration
Accessing Device Sensors: Accelerometer, Gyroscope
Location and Maps: GPS and Google Maps API (Basic)
Background Tasks: AsyncTask, WorkManager (Intro)
Permissions and Security

### Module 5:

App Deployment, Debugging and Project Development
Debugging Tools in Android Studio
Unit Testing and UI Testing Basics
App Signing and Versioning
Publishing App to Google Play (Overview)
Mini Project / Capstone Project
Best Practices in Android Development

## Labs / Practicals:

1. Create a “Hello World” Android app using Android Studio and test on emulator.
2. Design a login UI screen using different layout managers and widgets.
3. Create an app with multiple activities and pass data using intents.
4. Store user preferences using SharedPreferences and display them.
5. Develop a simple SQLite-based app for student record management.
6. Integrate Google Maps to display current location.
7. Capture image using device camera and display in the app.
8. Create a media player app that plays audio or video from the device.
9. Develop an app with sensor-based interaction, such as shake detection.
10. Final Mini Project: A complete Android app with at least one form of persistent data storage and sensor/location integration.

## References:

Kushwaha, D. S., & Misra, R.
Android Application Development: A Beginner's Guide
McGraw Hill Education, 2018.

Mednieks, Z., Dornin, L., Meike, G., & Nakamura, M.
Programming Android
O’Reilly Media, 3rd Edition, 2012.

Donn Felker
Android Application Development for Dummies
Wiley Publishing, 3rd Edition, 2015.

Joseph Annuzzi Jr., Lauren Darcey, and Shane Conder
Android Wireless Application Development (Volume I & II)
Addison-Wesley, 2015.

Wei-Meng Lee
Beginning Android Programming with Android Studio
Wiley, 4th Edition, 2021.

Online Resources

Android Developers Official Documentation
https://developer.android.com

Google Codelabs – Android Development Tutorials
https://developer.android.com/codelabs

Android Jetpack (UI, Room, WorkManager)
https://developer.android.com/jetpack

Kotlin Language Documentation
https://kotlinlang.org/docs/home.html

Udacity Course: Developing Android Apps with Kotlin
https://www.udacity.com/course/developing-android-apps-with-kotlin--ud9012

| Course Code | Course Name | L | T | P | Cr |
|---|---|---|---|---|---|
| DASH250050 | Entrepreneurship and Start-Ups | 3 | 1 | 0 | 4 |

### Course Outcomes:

CO1: Understand the fundamental concepts of entrepreneurship, including traits, motivation, business structures, and the role of entrepreneurs in economic development.

CO2: Identify, evaluate, and visualize innovative business ideas through effective market analysis and opportunity recognition.

CO3: Develop and implement business strategies related to marketing, competition analysis, risk management, and organizational structure.

CO4: Demonstrate knowledge of financial planning, start-up funding sources, investor relations, and intellectual property rights essential for small business management.

CO5: Design a comprehensive business plan that integrates strategic, financial, operational, and marketing perspectives for a successful start-up venture.

### Module 1:

Introduction to Entrepreneurship and Business Foundations
Definitions, Traits of an entrepreneur, Intrapreneurship, Motivation,Types of Business Structures, Similarities/differences between entrepreneurs and managers, Exit strategies for entrepreneurs, Bankruptcy, and Succession and harvesting strategy.

### Module 2:

Business Ideas and Visualization
Discovering ideas and visualizing the business, Activity map, Market Analysis: Identifying the target market.

### Module 3:

Business Strategy and Implementation
Competition evaluation and Strategy Development, Marketing and accounting, Risk analysis.

### Module 4:

Organizational and Financial Management
Company’s Organization Structure, Recruitment and management of talent, Financial organization and management.

### Module 5:

Start-up Financing, Investor Relations and IPR
Financing methods available for start-ups in India, Communication of Ideas to potential investors: Investor Pitch, Patenting and Licenses.

### Labs / Practicals:

N/A

### References:

1. The Startup Owner’s Manual: The Step-by-Step Guide for Building a Great Company, Steve Blank and Bob Dorf, K & S Ranch.
2. The Lean Startup: How Today’s Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses, Eric Ries Penguin, .
3. Demand: Creating What People Love Before They Know They Want It, Adrian J. Slywotzky with Karl Weber,

Headline Book Publishing .
4. The Innovator’s Dilemma: The Revolutionary Book That Will Change the Way You Do Business, Clayton M. Christensen, Harvard business .

| Course Code | Course Name | L | T | P | Cr |
|---|---|---|---|---|---|
| DCSA250039 | Introduction to IT Systems | 2 | 1 | 2 | 4 |

### Course Outcomes:

CO1. Comfortably work on Computer.
CO2. Install and Configure Operating System.
CO3. Assemble a PC and connect it to external devices.
CO4. Write documents, create worksheets, and prepare presentations.
CO5. Protect information and computers from basic abuses/attacks.

### Module 1:

Basic Internet skills and hardware components: Understanding browser, efficient use of search engines, awareness about Digital India portals (state and national portals) and college portals.
General understanding of various computer hardware components – CPU, Memory, Display, Keyboard, Mouse, HDD and other Peripheral Devices.

### Module 2:

Operating System Installation and shell commands:OS Installation (Linux and MS Windows), Unix Shell and Commands, vi editor

### Module 3:

HTML5,CSS and personal webpage design: HTML5, CSS, making basic personal webpage

### Module 4:

Office Tools: OpenOffice Writer, OpenOffice Spreadsheet (Calc), OpenOffice Impress

### Module 5:

Information security: Introduction to Computer Security, Types of Security Threats, Security Measures, Information security best practices

### Labs / Practicals:

1. Identify various computer components.
2. Understand PC assembly basics.
3. Learn OS installation procedure.
4. Basic BIOS/UEFI Configuration
5. Installing and Using Office Productivity Tools
6. Introduction to Networking Basics
7. Internet and Web Browsing
8. Installing and Uninstalling Software

### References:

1. R.S. Salaria, Computer Fundamentals, Khanna Publishing House
2. Ramesh Bangia, PC Software Made Easy – The PC Course Kit, Khanna Publishing House
3. Mastering Linux Shell Scripting: A practical guide to Linux command-line, Bash scripting, and Shell programming, by Mokhtar Ebrahim, Andrew Mallett

| Course Code | Course Name | L | T | P | Cr |
|---|---|---|---|---|---|
| DCSE250078 | Fundamentals of ICT | 3 | 0 | 2 | 4 |

### Course Outcomes:

CO1. Use computer system and its peripherals for given purpose
CO2. Understand and evaluate various input, output, and storage devices, along with data types in ICT, including their functions, applications, advantages, and limitations.
CO3. Prepare Business document using Word Processing Tool
CO4. Explain the principles of computer networks, internet, and wireless communication technologies.
CO5. Use different types of Web Browsers and Apps

### Module 1:

Definition ICT, Salient Features of ICT, Main theme of ICT, Components of ICT, Effectiveness of ICT in Modern educational System and research program.

### Module 2:

Introduction to Computer System: Input Devices like keyboards, numeric keypads, pointing devices (including mouse, touch pad and tracker ball), remote controls, joysticks, touch screens, magnetic stripe readers, chip readers, PIN pads, scanners, digital cameras, microphones, sensors, graphics tablet, MICR, OMR, OCR, barcode readers, video cameras, web cams, light pens and suitable uses of the input devices stating the advantages and disadvantages of each, Output Devices like monitors (CRT, TFT), projectors, printers (laser, ink jet and dot matrix), plotters, speakers, control devices and suitable uses of the input devices stating the advantages and disadvantages of each, Storage devices and media and their effective uses in ICT, Data types used in ICT.

### Module 3:

Word Processing
Word Processing: Overview of Word processor Basics of Font type, size, colour, Effects like Bold, italic, underline, Subscript and superscript, Case changing options, Previewing a document, Saving a document, Closing a document and exiting application.
Editing a Document: Navigate through a document, Scroll through text, Insert and delete text, Select text, Undo and redo commands, Use drag and drop to move text, Copy, cut and paste, Use the clipboard, Clear formatting, Format and align text, Formatting
Changing the Layout of a Document: Adjust page margins, Change page orientation, Create headers and footers, Set and change indentations, Insert and clear tabs
Inserting Elements to Word Documents: Insert and delete a page break, Insert page numbers, Insert the date and time, Insert special characters (symbols), Insert a picture from a file, Resize and reposition a picture
Working with Tables: Insert a table, Convert a table to text, Navigate and select text in a table, Resize table cells, Align text in a table, Format a table, Insert and delete columns and rows, Borders and shading, Repeat table headings on subsequent page
Working with Columned Layouts and Section Breaks: a Columns, Section breaks, Creating columns, Newsletter style columns, Changing part of a document layout or formatting, Remove section break, Add columns to remainder of a document, Column widths, Adjust

### Module 4:

Basics of Networking, Types of Networking, Introduction to network devices, sharing of computer resources, creating a local network, Introduction to Internet and its features, Different procedures for accessing the Internet, Server-Client architecture, Concept of Remote networking:Remote server, Remote host.

### Module 5:

The ways in which ICT is used, Systems analysis and design for ICT, Practical tests and their Assessment criteria, Data analysis, Procedures for conducting practical tests for ICT related projects, Grade descriptions and Security issues.

### Labs / Practicals:

Word Processing:
Creating and formatting documents using software like Microsoft Word.
Applying styles, page numbers, and headers/footers.
Mail merging to create personalised documents.
Data Manipulation:
Working with databases using software like Microsoft Access.
Importing and exporting data.
Sorting, filtering, and querying data.
Using formulas and functions in spreadsheets (e.g., Excel).
Presentation Authoring:
Creating presentations using software like Microsoft PowerPoint.
Designing slides with appropriate content, images, and transitions.
Adding footers, headers, and automatic date updates.
Internet and Networking:
Understanding and using web browsers (e.g., Chrome, Firefox).
Searching for information online.
Understanding basic networking concepts.
Using email and other communication tools.

### References:

Text Books:
1. Frank Webster,“Theories of the Information Society”, 4th Edition, International Library of Sociology. 2. “Complete ICT for Cambridge IGCSE 2nd edition”, OXFORD References:
3. Anita Goel ,”Computer Fundamentals “, Pearson
4. P. K. Sinha, “Fundamental of Computers”, B.P.B. Publications

| Course Code | Course Name | L | T | P | Cr |
|---|---|---|---|---|---|
| DCSA250001 | Fundamentals of Image and Video Processing | 2 | 0 | 4 | 4 |

### Course Outcomes:

CO1. Understand the basic principles of image and video representation.
CO2. Apply enhancement and restoration techniques on digital images.
CO3. Implement image compression and segmentation algorithms.
CO4. Perform morphological operations and basic object recognition.
CO5. Analyze motion in video sequences and understand basic video compression.

### Module 1:

Introduction to Image and Video Processing: Digital image and video fundamentals, image and video representation, color models, sampling, quantization, and digitization.

### Module 2:

Image Enhancement and Restoration: Histogram equalization, contrast stretching, spatial filtering, frequency domain filtering, and noise models.

### Module 3:

Image Compression and Segmentation: Predictive coding, transform coding (DCT, DWT), region-based and edge-based segmentation, clustering and thresholding techniques.

### Module 4:

Morphological Operations and Object Recognition: Dilation, erosion, opening, closing, boundary extraction, region filling, connected components, feature extraction.

### Module 5:

Video Processing and Motion Estimation: Frame difference, background subtraction, optical flow, motion models, video compression basics (MPEG), applications.

### Labs / Practicals:

1. Perform grayscale and color image display using a programming tool (e.g., Python/Matlab).
2. Apply histogram equalization and contrast enhancement on sample images.
3. Implement spatial and frequency domain filters (mean, Gaussian, high-pass).
4. Apply noise removal techniques on noisy images.
5. Perform image compression using DCT and observe compression artifacts.
6. Segment images using thresholding and edge-based techniques.
7. Demonstrate morphological operations like erosion and dilation.
8. Extract features such as edges, corners, and shapes from images.
9. Perform frame differencing and background subtraction in a video sequence.
10. Visualize optical flow in motion sequences using basic motion estimation.

### References:

Textbooks:
1. Rafael C. Gonzalez and Richard E. Woods, “Digital Image Processing”, 4th Edition, Pearson Education, 2018.
2. Alan C. Bovik, “The Essential Guide to Video Processing”, Academic Press, 2009.
3. Anil K. Jain, “Fundamentals of Digital Image Processing”, Prentice Hall, 1989.

References / Online Resources
1. https://nptel.ac.in/courses/117105079

2. https://homepages.inf.ed.ac.uk/rbf/CVonline/
3. https://scikit-image.org/
4. https://docs.opencv.org/
5. https://www.imageprocessingplace.com/

| Course Code | Course Name | L | T | P | Cr |
|---|---|---|---|---|---|
| DCSE250029 | Computer Graphics | 3 | 1 | 0 | 4 |

### Course Outcomes:

CO1: Understand the basics of computer graphics systems and their architecture. Discuss various algorithms for scan conversion and filling of basic objects and their comparative analysis.
CO2: Apply geometric transformations and model graphical objects using OpenGL.
CO3: Develop interactive applications with input and event handling.
CO4: Illustrate concepts of 3D viewing, projections, and shading models.
CO5: Implement algorithms for clipping, rasterization, and rendering pipeline stages.

### Module 1:

Introduction to Computer Graphics and OpenGL
Introduction to Computer Graphics: Applications, physical vs. synthetic images,
Graphics Systems: Imaging systems, synthetic camera model, Graphics Architectures: Programmable pipelines, performance metrics, Graphics Programming: Sierpinski gasket, 2D applications, OpenGL Basics: OpenGL API, primitives, attributes, colors Viewing & Control: Viewing, control functions, recursive graphics, 3D Gasket & Implicit Functions

### Module 2:

Interaction and Input Mechanisms
Interaction & Input Devices: Event handling, input models, Client-Server Architecture, Display Lists and Modeling: Reusability, efficiency, Event Driven Programming: Menus, picking, Building Interactive Models: CAD systems, Animation and Logic Operations

### Module 3:

Geometric Transformations and Modeling
Geometric Objects: Scalars, points, vectors, Three-dimensional Primitives; Coordinate Systems and Frames; Modeling a Colored Cube; Affine Transformations; Rotation, Translation and Scaling; Geometric Objects and Transformations; Transformation in Homogeneous Coordinates; Concatenation of Transformations; OpenGL Transformation Matrices; Interfaces to 3D applications; Quaternion's.

### Module 4:

Viewing and Shading
Classical and computer viewing; Viewing with a Computer; Positioning of the camera; Simple projections; Projections in OpenGL; Hidden- surface removal; Interactive Mesh Displays; Parallel-projection matrices; Perspective-projection matrices; Projections and Shadows. Light and Matter; Light Sources; The Phong Lighting model; Computation of vectors; Polygonal Shading; Approximation of a sphere by recursive subdivisions; Light sources in OpenGL; Specification of materials in OpenGL; Shading of the sphere model; Global Illumination.

### Module 5:

Implementation Techniques and Rendering Pipeline
Basic Implementation Strategies; Four major tasks; Clipping; Line-segment clipping; Polygon clipping; Clipping of other primitives; Clipping in three dimensions; Rasterization; Bresenham's algorithm; Polygon Rasterization; Hidden-surface removal; Antialiasing; Display considerations.

### Labs / Practicals:

NA

### References:

1.Edward Angel: Interactive Computer Graphics A Top-Down Approach with OpenGL, 5th Edition, Pearson Education, 2008.
2.Donald Hearn and Pauline Baker: Computer Graphics OpenGL Version 3th Edition, Pearson Education, 2004.
3.F.S. Hill Jr.: Computer Graphics Using OpenGL, 3th Edition, PHI, 2909.
4.James D Foley, Andries Van Dam, Steven K Feiner, John F Hughes, Computer Graphics, Pearson Education 1997.

| Course Code | Course Name | L | T | P | Cr |
|---|---|---|---|---|---|
| DCSA250065 | Programming in Java | 3 | 1 | 0 | 4 |

### Course Outcomes:

CO1. Understand the fundamental concepts of JAVA.
CO2. Understand the fundamental concepts of object-oriented programming (OOP)
CO3. Implement object-oriented programs using classes and methods
CO4. Apply exception handling and file I/O in Java programs.
CO5. Implementation of JDBC.

### Module 1:

Introduction to Java: History and evolution of Java, Features of Java (Platform independence, OOP, robustness, etc.),Java Virtual Machine (JVM), JDK, and JREJava program structure, Java development tools (IDE: NetBeans, Eclipse, or IntelliJ IDEA),Data types, variables, Keywords and identifiers, Input/Output, Operators,Decision-making statements, Looping statements, Jump statements, Array

### Module 2:

Introduction to Object-Oriented Programming: Principles of OOPs, Classes and Objects in Java, Access Modifiers and Static Members, Constructors, Method Overloading, Static Members, Garbage Collection

### Module 3:

Inheritance, Interfaces, and Packages: Inheritance, Polymorphism, Abstract Classes and Methods, Interfaces, Packages

### Module 4:

Exception Handling in Java: Introduction to Exceptions, Exception Hierarchy, Handling Exceptions, Throwing Exceptions, User-Defined Exceptions, Best Practices in Exception Handling

### Module 5:

Java Database Connectivity: Introduction to JDBC, Establishing a Database Connection, Executing SQL Queries, Handling Exceptions and Transactions

### Labs / Practicals:

NA

### References:

1. The Complete Reference / Herbert Schildt / Tata Mc-Graw Hill Pub. Co. Ltd. Head First Java / K. Sierra & B. Bates , O'Reilly
2. Internet and Java Programming / R. Krishnamoorthy & S. Prabhu / New Age
3. Java foundations of programming/ Prentice Hall of India Pvt. Ltd.
International (p) Ltd
4. Teach yourself Web Technologies Part – I / Ivan Bayross / BPB Publications

| Course Code | Course Name | L | T | P | Cr |
|---|---|---|---|---|---|
| DOAI250009 | Big Data Analytics | 3 | 0 | 2 | 4 |

### Course Outcomes:

CO1: Describe big data characteristics, challenges, and applications across industries and domains.

CO2: Evaluate and apply NoSQL databases and distributed storage architectures for big data management.

CO3: Install, configure, and operate Hadoop and HDFS for batch-oriented big data processing.

CO4: Develop and implement data analysis workflows using Hadoop ecosystem tools including Hive, Pig, and HBase.

CO5: Design and implement real-time data processing and analytics using Spark Streaming and PySpark.

CO6: Apply visualization and regression techniques for analytical insights from large-scale data.

### Module 1:

Introduction to Big Data Ecosystem

Big Data Characteristics: Volume, Variety, Velocity, Veracity, and Value, Business applications and case studies of big data analytics, Challenges in conventional systems, Big Data Analytics Lifecycle, Tools and Technologies in Big Data, Hadoop vs. Traditional Systems, NoSQL Overview (Key-Value, Document, Columnar, Graph databases), Basics of MongoDB and Cassandra for big data storage.

### Module 2:

Hadoop Framework for Big Data

History and Evolution of Hadoop, Hadoop Distributed File System (HDFS): Architecture, Internals, and Operations, Block-level Storage, Fault Tolerance, MapReduce Paradigm: Concepts, Anatomy of a MapReduce Job, Failures, Scheduling, Shuffle, Sort, and Task Execution, Input / Output Formats in MapReduce, Hadoop Streaming and Integrations, Developing MapReduce Applications with Java and Python APIs.

### Module 3:

Data Analysis with Hadoop Ecosystem

Hive: Data Modeling, Data Types, File Formats, HiveQL (DDL, DML, DQL), Querying Structured Data, Hive Services and Optimization.
Pig: Data Flow Language, Pig Latin Programming, Grunt Shell, Data Models, Operators, Joins, Developing and Testing Pig Scripts.
HBase: Architecture, Data Model, Integration with Hadoop, Use Cases.
Introduction to Data Lakes using Hadoop.
Hands-on with Hadoop Ecosystem for ETL pipelines.

### Module 4:

Real-Time Big Data Analytics and Streaming

Stream Processing Fundamentals, Stream Data Models, Stream Architecture, Concepts of Sampling, Filtering, Counting in Streams, Real-time Sentiment Analysis, Stock Market Analysis using Streams, Introduction to Apache Kafka for Streaming Data Ingestion, Spark Basics: RDD, DataFrames, Spark SQL, In-Memory Computing, PySpark APIs, Spark Streaming for Real-Time Processing, Real-Time Analytics Applications, Case

Studies.

### Module 5:

Analytical Models and Visualization for Big Data

Regression Techniques: Simple Linear Regression, Multiple Linear Regression, Model Interpretation for Large-Scale Data, Feature Engineering for Big Data, Visualization Techniques: Interactive Dashboards, Graphical Tools for Big Data (Tableau / PowerBI concepts), Visual Analytics: Trends, Patterns, Correlations, Interaction Models, Real-World Applications of Big Data Visualizations.

### Labs / Practicals:

1. Setting up Hadoop, HDFS, and YARN on local and cloud-based environments (AWS / Azure / GCP).

2. Developing basic MapReduce jobs using Java and Python.

3. Implementing ETL pipelines using Hive and Pig.

4. Performing analytics with HBase and Hive.

5. Real-time data streaming with Apache Kafka and Spark Streaming (using PySpark).

6. Implementing basic ML pipelines over Spark (MLlib).

7. Visualizing Big Data using interactive tools (Matplotlib, Plotly, Tableau / PowerBI).

8. Group Project: Building and deploying an end-to-end Big Data Analytics pipeline on a real-world dataset.

### References:

1. Tom White, Hadoop: The Definitive Guide, 4th Edition, O'Reilly Media, 2015.

2. Anand Rajaraman, Jeffrey David Ullman, Mining of Massive Datasets, Cambridge University Press, 3rd Edition, 2020.

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

4. Jiawei Han, Micheline Kamber, Jian Pei, Data Mining: Concepts and Techniques, 3rd Edition, Morgan Kaufmann / Elsevier, 2011.

5. Holden Karau, Andy Konwinski, Patrick Wendell, Matei Zaharia, Learning Spark: Lightning-Fast Big Data Analysis, 2nd Edition, O'Reilly Media, 2020.

6. Paul Zikopoulos, Dirk deRoos, Krishnan Parasuraman, Thomas Deutsch, James Giles, David Corrigan, Harness the Power of Big Data – The IBM Big Data Platform, Tata McGraw Hill, 2012.

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

8. Da Ruan, Guoqing Chen, Etienne E. Kerre, Geert Wets, Intelligent Data Mining, Springer, 2007.

Official Documentation (Highly Recommended for Labs / Practical Work):

9. Official documentation for Hadoop: https://hadoop.apache.org/

10. Official documentation for Spark: https://spark.apache.org/docs/latest/

11. Official documentation for Hive: https://cwiki.apache.org/confluence/display/Hive/Home

12. Official documentation for Pig: https://pig.apache.org/

13. Official documentation for HBase: https://hbase.apache.org/

14. Official documentation for Kafka: https://kafka.apache.org/documentation/

15. Official documentation for PySpark: https://spark.apache.org/docs/latest/api/python/

| Course Code | Course Name | L | T | P | Cr |
|---|---|---|---|---|---|
| DCSA250055 | Multimedia Technologies | 3 | 0 | 2 | 4 |

### Course Outcomes:

CO1. Analyze multimedia components such as text, graphics, audio, video, and animation.
CO2. Create and design video and audio effects using multimedia tools.
CO3. Understand multimedia formats, standards, and their properties.
CO4. Develop animated sequences and apply basic principles of animation.
CO5. Design multimedia applications using appropriate tools for practical scenarios.

### Module 1:

Introduction to Multimedia:
Definition, basics, elements, applications, multimedia workstation architecture. Hypermedia documents, hypertext, hyperspeech, 3D technologies, holography (definitions only). Overview of multimedia software tools, open-source replacements, multimedia OS.

### Module 2:

Defining Objects for Multimedia Systems:
Text, images, audio, video; multimedia data interface and processing standards. Bitmaps, vector graphics, rendering, color palettes, digital audio, MIDI, sound integration, animation principles and techniques, video formats and converters.

### Module 3:

Multimedia Data and Standards:
Compression: lossy/lossless concepts, JPEG, video and audio standards. File format standards: RTF, RIFF, GIF, PNG, TIFF, MIDI, JPEG, AVI, WAV, BMP, WMF, MPEG (list only).

### Module 4:

Multimedia Devices and Making Multimedia:
Input/output technologies, multimedia project stages. Text editing, OCR, painting, 3D modeling, sound/video editing, authoring systems and tools.

### Module 5:

Multimedia and Internet:
Internet bandwidth bottleneck, multimedia services, plug-ins, delivery vehicles. Wireless and mobile multimedia networks, streaming architecture, webcasting, on-demand services, video conferencing.

### Labs / Practicals:

1. Use HTML to play audio/video formats on desktop/mobile browsers
2. Import and manipulate images on a workspace
3. Create a notebook wrapper or invitation using suitable software
4. Use a scanner and stitch images into a panorama
5. Edit video clips using software (trim, rotate, merge, add effects)
6. Create a 2D animation using motion guide layer and masking
7. Animate a moving cloud using animation software
8. Develop an animated web page with sound using HTML editor

9. Compress/decompress audio/video and convert file formats
10. Create a pencil sketch from an image using software
11. Morph and overlap two pictures using editing tools
12. Design a certificate using different text effects
13. Record, edit, split, and merge audio from different sources
14. Create 3D metallic text using animation tools
15. Design an innovative institute logo incorporating key features

## References:

1. Ze-Nian Li and M. S. Drew, Fundamentals of Multimedia, 2nd Edition, Pearson Education, 2014
2. Tay Vaughan, Multimedia: Making It Work, 8th Edition, Tata McGraw-Hill, 2017
3. Ralf Steinmetz and Klara Nahrstedt, Multimedia Computing, Communication and Applications, Pearson Education, 2012

| Course Code | Course Name | L | T | P | Cr |
|---|---|---|---|---|---|
| DCSE250020 | Cloud Computing | 3 | 0 | 2 | 4 |

### Course Outcomes:

CO1: Explain the evolution, architecture, and key characteristics of cloud computing, including service models (IaaS, PaaS, SaaS) and deployment types (public, private, hybrid).

CO2: Describe various types and implementation levels of virtualization, including CPU, memory, storage, and network virtualization in cloud environments.

CO3: Analyze layered cloud architecture and address design challenges related to inter-cloud resource management and platform deployment.

CO4: Apply distributed and parallel programming paradigms such as MapReduce and Hadoop for cloud-based application development and understand various cloud platforms like AWS, OpenStack, and Google App Engine.

CO5: Evaluate cloud security challenges and solutions, including data and application security, VM security, and risk management strategies in cloud environments.

### Module 1:

Introduction to Cloud Computing

Evolution of Cloud Computing, System Models for Distributed and Cloud Computing, NIST Cloud Computing Reference Architecture, Features of Cloud Computing, Cloud Services – IaaS, PaaS, SaaS, Cloud service Providers – Public , Private and Hybrid Clouds.

### Module 2:

Introduction to component virtualization

Basics of Virtualization, Types of Virtualization, Implementation Levels of Virtualization, Virtualization of CPU, Memory, I/O Devices, Desktop Virtualization, Server Virtualization, Storage Virtualization, Network Virtualization.

### Module 3:

Architectural Design of Compute and Storage Clouds

Layered Cloud Architecture Development, Design Challenges, Inter Cloud Resource Management, Resource Provisioning and Platform Deployment, Global Exchange of Cloud Resources.

### Module 4:

Parallel and Distributed Programming Paradigms

Map Reduce, Twister and Iterative MapReduce, Hadoop Library from Apache, Mapping Applications, Programming Support, Google App Engine, Amazon AWS, Cloud Software Environments - Eucalyptus, Open Nebula, OpenStack.

### Module 5:

Security Overview

Cloud Security Challenges, Software-as-a-Service Security, Security Governance, Risk Management, Security Monitoring, Security Architecture Design, Data Security, Application Security, Virtual Machine Security.

### Labs / Practicals:

1. Exploring Cloud Deployment and Service Models
2. Simulating Public, Private, and Hybrid Clouds
3. Virtualization of CPU and Memory
4. Network and Storage Virtualization
5. Deployment of a Layered Cloud Architecture
6. Resource Provisioning and Scaling
7. Deploying a Web Application
8. Simulating Security Attacks and Defenses in a Cloud VM
9. Implementing Access Control and Encryption on Cloud Storage

### References:

[1] R. Buyya, C. Vecchiola, and T. Selvi, Mastering Cloud Computing: Foundations and Applications Programming. New Delhi, India: McGraw-Hill Education, 2013.
[2] A. T. Velte, T. J. Velte, and R. Elsenpeter, Cloud Computing: A Practical Approach. New Delhi, India: McGraw-Hill Education, 2010.
[3] K. Hwang, G. C. Fox, and J. J. Dongarra, Distributed and Cloud Computing: From Parallel Processing to the Internet of Things. Burlington, MA, USA: Morgan Kaufmann, 2012.
[4] T. Erl, R. Puttini, and Z. Mahmood, Cloud Computing: Concepts, Technology & Architecture. Boston, MA, USA: Pearson Education, 2013.
[5] G. Reese, Cloud Application Architectures: Building Applications and Infrastructure in the Cloud. Sebastopol, CA, USA: O'Reilly Media, 2009.