



Imperial College of Engineering

Affiliated to Tribhuvan University · Bhaktapur, Nepal

COURSE SYLLABUS HANDBOOK

Bachelor in Computer Engineering

BCT · Four years · Eight semesters

The official curriculum of the Institute of Engineering, Tribhuvan University, for students admitted in 2080 and later. Years I to III, 36 course syllabi.



BCT · 2080 CURRICULUM

About this handbook

This handbook brings every published syllabus for the Bachelor in Computer Engineering together in one place, arranged semester by semester, so that a student can see the whole degree from the first class to the last.

Each syllabus page is reproduced unchanged from the Institute of Engineering (IOE), Tribhuvan University. IOE writes and revises the curriculum and remains its authority. As a TU-affiliated college, Imperial College of Engineering teaches this same curriculum, and our students sit the same IOE examinations and receive the same Tribhuvan University degree as students on every other campus.

Years I to III are complete. IOE has named the Year IV courses but has not yet published their syllabi. They are listed on page 159 and will be added when IOE releases them.

DEGREE	Bachelor in Computer Engineering (BCT)
UNIVERSITY	Tribhuvan University, Institute of Engineering
CURRICULUM	2080 batch and later
DURATION	Four years, eight semesters
IN THIS BOOK	36 syllabi, Semesters 1 to 6
SOURCE	ioe.tu.edu.np, compiled 30 September 2026



Contents

Semester 1	YEAR I, PART I	5
ENSH 101	Engineering Mathematics I	6
ENCT 101	Computer Programming	8
ENME 101	Engineering Drawing	12
ENEX 101	Fundamental of Electrical and Electronics Engineering	16
ENSH 102	Engineering Physics	19
ENME 106	Engineering Workshop	25
Semester 2	YEAR I, PART II	28
ENSH 151	Engineering Mathematics II	29
ENCT 151	Object Oriented Programming	32
ENEX 152	Digital Logic	36
ENEX 151	Electronic Device and Circuits	40
ENSH 153	Engineering Chemistry	43
ENEE 154	Electrical Circuits and Machines	48
Semester 3	YEAR II, PART I	53
ENSH 201	Engineering Mathematics III	54
ENSH 204	Communication English	57
ENCT 201	Computer Graphics and Visualization	60
ENCT 202	Foundation of Data Science	64
ENCT 203	Theory of Computation	68
ENEX 201	Microprocessors	71



Semester 4	YEAR II, PART II	75
ENSH 252	Numerical Methods	76
ENEX 252	Instrumentation	80
ENEX 254	Electromagnetics	84
ENCT 252	Data Structure and Algorithm	87
ENCT 253	Data Communication	91
ENCT 254	Operating System	94
Semester 5	YEAR III, PART I	97
ENSH 304	Probability and Statistics	98
ENCT 301	Database Management System	101
ENCT 302	Web Application Programming	105
ENCT 303	Computer Organization and Architecture	109
ENCT 304	Computer Networks	113
ENCT 325-344	Elective I	117
Semester 6	YEAR III, PART II	129
ENCE 356	Engineering Economics	130
ENCT 351	Artificial Intelligence	133
ENCT 352	Software Engineering	137
ENCT 353	Simulation and Modeling	141
ENCT 354	Minor Project	145
ENCT 385-399	Elective II	147
Year IV	SYLLABI NOT YET PUBLISHED	159



YEAR I, PART I

Semester 1

6 COURSES IN THIS SEMESTER

ENSH 101	Engineering Mathematics I	6
ENCT 101	Computer Programming	8
ENME 101	Engineering Drawing	12
ENEX 101	Fundamental of Electrical and Electronics Engineering	16
ENSH 102	Engineering Physics	19
ENME 106	Engineering Workshop	25



ENGINEERING MATHEMATICS I

ENSH 101

Lecture : 3
Tutorial : 2
Practical : 0

Year : I
Part : I

Course Objectives:

To equip the students with the essential mathematical skills and techniques that are relevant to the engineering fields and enable them to solve engineering problems using mathematical methods.

1 Derivatives and its Applications (10 hours)

- 1.1 Review of derivative and differentiability, mean value theorems with interpretations
- 1.2 Indeterminate forms, types and their real life examples, L-Hospital's rule
- 1.3 Power series of single valued functions
 - 1.3.1 Taylor's series
 - 1.3.2 Maclaurin's series
- 1.4 Asymptotes to Cartesian and Polar curves
- 1.5 Pedal equation to Cartesian and Polar curves
- 1.6 Curvature and radius of curvature for Cartesian curves

2 Antiderivatives and its Applications (11 hours)

- 2.1 Review of definite and indefinite integrals
- 2.2 Differentiation under integral sign
- 2.3 Improper integrals
- 2.4 Application of Beta and Gamma functions
- 2.5 Area, arc length, volume and surface of revolution in plane for Cartesian curves
- 2.6 Centroid and moment of inertia under area of curve

3 Ordinary Differential Equations and its Applications (10 hours)

- 3.1 Review of order, degree, solution of first order first degree differential equations by variable separation method and solution of homogeneous equations
- 3.2 Linear differential equation and equations reducible to linear differential equation of first order Bernoulli's equation, modeling electric circuit
- 3.3 First order and higher degree differential equations; Clairaut's form

- 3.4 Linear second order differential equations with constant coefficient and variable coefficients reducible to constant coefficients, Cauchy's equations and modeling mass spring system
- 3.5 Application in physical sciences and engineering

4 Plane Analytic Geometry (4 hours)

- 4.1 Transformation of coordinates: Translation and rotation
- 4.2 Equation of conic in Cartesian and Polar form, identification of conics

5 Three dimensional geometry (10 hours)

- 5.1 The straight line: Symmetrical and general form
- 5.2 Coplanar lines
- 5.3 Shortest distance
- 5.4 Sphere: General equation, plane section by planes, tangent planes
- 5.5 Introduction to right circular cone and right circular cylinder

Tutorials (30 hours)

- 1. Derivatives and its applications
- 2. Antiderivatives and its applications
- 3. Ordinary differential equations and its applications
- 4. Plane analytic geometry
- 5. Three dimensional geometry

Reference

- 1. Jeffery, A., (2001), Advanced Engineering Mathematics. Academic Press.
- 2. O'Neill, P.V., (2003), Advanced Engineering Mathematics. Thomson Learning.
- 3. Kreyszig, A. (1993), Advanced engineering Mathematics (Latest Edition). John Wiley & Sons.
- 4. Sastry, S.S. (2008), Engineering Mathematics Volume I and II. PHI India.
- 5. Wylie, C., Barrett, L. (1995), Advanced Engineering Mathematics (Latest Edition), McGraw-Hill College.
- 6. Thomas, T., Finny, R. (1984), Calculus and Analytic Geometry (Latest Edition.), Addison-Wesley.

COMPUTER PROGRAMMING

ENCT 101

Lecture : 3

Tutorial : 1

Practical : 3

Year : I

Part : I

Course Objectives:

The primary goal of this course is to provide students with a solid foundation in the principles of programming and to impart practical skills in the C programming language. This course ensures that students comprehend the fundamental concepts of variables, data types, control structures, and functions within the context of C. Advanced topics such as pointers, structures, file handling and the standard C library are explored to broaden students' programming capabilities. Also, through project-based assessments and evaluations, students apply their knowledge to real-world scenarios, fostering creativity and project development skills.

1 Introduction to Computer Programming

(3 hours)

- 1.1 Definition of a computer program and programming language
- 1.2 Types and generations of programming languages
- 1.3 Problem-solving using a computer
 - 1.3.1 Problem analysis
 - 1.3.2 Algorithm and flowchart
 - 1.3.3 Programming
 - 1.3.4 Compilation, linking and execution
 - 1.3.5 Debugging and testing
 - 1.3.6 Documentation

2 Overview of C Programming

(3 hours)

- 2.1 Introduction to C programming
- 2.2 History and importance of C
- 2.3 C headers and library functions
- 2.4 Basic structure of a C program
- 2.5 Preprocessor directives
- 2.6 Tokens in C (Character set, keywords and identifiers)
- 2.7 Type casting (Implicit and explicit)
- 2.8 Data types, variables and constants
- 2.9 Compiler and IDE for C programming

3 Operators and Expressions (4 hours)

- 3.1 Introduction to operators and expressions
- 3.2 Arithmetic, relational and logical operators
- 3.3 Assignment, increment and decrement operators
- 3.4 Conditional, bitwise and special operators
- 3.5 Comma operator, size of operator
- 3.6 Evaluation and type conversion in expressions
- 3.7 Operator precedence and associativity

4 Input and Output (3 hours)

- 4.1 Introduction to data I/O in C
- 4.2 Unformatted I/O
 - 4.2.1 Character I/O
 - 4.2.2 String I/O
- 4.3 Formatted I/O
 - 4.3.1 Control string (Flags, field width, precision, and specifier)
 - 4.3.2 Formatted I/O (scanf(), printf())

5 Control Structures (8 hours)

- 5.1 Introduction to simple and compound statement
- 5.2 Sequential statement
- 5.3 Branching statement
 - 5.3.1 Simple if statement
 - 5.3.2 if-else statement
 - 5.3.3 Nested if-else statement
 - 5.3.4 else-if ladder
 - 5.3.5 switch statement
 - 5.3.6 go to statement
- 5.4 Looping statement
 - 5.4.1 for loop
 - 5.4.2 while loop
 - 5.4.3 do while
 - 5.4.4 Nested loop
- 5.5 Loop interruption
 - 5.5.1 break
 - 5.5.2 continue

6 Array and Pointer (7 hours)

- 6.1 Introduction to an array
- 6.2 One-dimensional array
- 6.3 Two-dimensional array

- 6.4 Multidimensional array
- 6.5 Introduction to string
- 6.6 String handling functions
- 6.7 Definition of a pointer
- 6.8 Pointer declaration
- 6.9 Pointer arithmetic
- 6.10 Relationship between pointer and arrays

7 User-defined Functions

(6 hours)

- 7.1 Introduction to function
- 7.2 Advantages of function
- 7.3 Elements of user-defined function
 - 7.3.1 Function definition
 - 7.3.2 Function prototype
 - 7.3.3 Function parameters
- 7.4 Storage class
- 7.5 Scope rules
- 7.6 Category of functions
 - 7.6.1 Functions with no arguments and no return values
 - 7.6.2 Functions with arguments and no return values
 - 7.6.3 Functions with arguments and return values
 - 7.6.4 Functions with no arguments and return values
- 7.7 Recursive functions
- 7.8 Function call by values and reference
- 7.9 Passing array and string to function

8 Structures

(5 hours)

- 8.1 Defining a structure
- 8.2 Declaring and accessing structure elements
- 8.3 Initializing structure
- 8.4 Array of structure
- 8.5 Array as member to structure
- 8.6 Pointer as member to structure
- 8.7 Structure as a member to structure
- 8.8 Passing and returning structures to/from function

9 File Management

(4 hours)

- 9.1 Introduction
- 9.2 Binary and text file in C
- 9.3 File opening modes
- 9.4 Defining, opening and closing file

- 9.5 Input-output operations on files
 - 9.5.1 Character I/O (fputc(), fgetc())
 - 9.5.2 String I/O (fgets(), fputs())
 - 9.5.3 Formatted I/O (fscanf(), fprintf())
 - 9.5.4 Record I/O (fwrite(), fread())
- 9.6 Overview of random file access
- 9.7 Error handling

10 Recent Trends in Programming

(2 hours)

- 10.1 Introduction to object oriented programming (OOP)
- 10.2 Definitions of class, method and object in OOP
- 10.3 Difference between procedure oriented and OOP
- 10.4 Overview of other high level programming languages

Practical

(45 hours)

- 1. Lab 1: Introduction and demonstrations of projects written in C
- 2. Lab 2: Formatted and unformatted Input/output in C
- 3. Lab 3: Branching in control structure
- 4. Lab 4: Looping in control structure
- 5. Lab 5: Array in C
- 6. Lab 6: String in C
- 7. Lab 7: Pointers in C
- 8. Lab 8: User defined functions in C
- 9. Lab 9: Structure in C
- 10. Lab 10: File handling in C
- 11. Group project on C maximum 4 students in a group at the end of the course.

Reference

- 1. Lafore, R. (1993). *The Waite Group's C programming using Turbo C++* (1st ed.). SAMS Publishing.
- 2. Balagurusamy, E. (2019). *Programming in ANSI C* (8th ed.). McGraw Hill Education.
- 3. Gottfried, B. S. (2010). *Programming with C* (3rd ed.). McGraw Hill Education.

ENGINEERING DRAWING

ENME 101

Lecture : 2
Tutorial : 0
Practical : 4

Year : I
Part : I

Course Objectives:

To develop basic projection concepts with reference to points, lines, planes and geometrical solids. Also, to develop sketching and drafting skills to facilitate communication.

1 Instrumental Drawing, Technical Lettering Practices and Techniques

(1 hour)

- 1.1 Equipment, materials and drawing sheets
- 1.2 Description of drawing instruments, auxiliary equipment and drawing materials
- 1.3 Techniques of instrumental drawing
- 1.4 Pencil sharpening, securing paper, proper use of T- squares, triangles, scales dividers, compasses, erasing shields, French curves, inking pens
- 1.5 Line: Types and uses, thickness

2 Dimensioning

(1 hour)

- 2.1 Fundamentals and techniques
- 2.2 Size and location dimensioning, SI conversions
- 2.3 Scales: Types and representative factor
- 2.4 Use of scales, measurement units, reducing and enlarging drawings
- 2.5 Placement of dimensions: Aligned and unidirectional, chain, parallel/baseline and combined type
- 2.6 Tolerance dimensioning

3 Geometrical Construction

(2 hours)

- 3.1 Plane geometrical construction: Proportional division of lines, trisection of angles, smooth arc and line tangents
- 3.2 Methods for drawing regular polygons and standard curves such as ellipses, parabolas, hyperbolas, involutes, spirals, cycloids and helices (Cylindrical and conical), ogee curve
- 3.3 Techniques to reproduce a given drawing (By construction)

4 Basic Descriptive Geometry

(4 hours)

- 4.1 Introduction to orthographic projection, principal planes, four quadrants or angles
- 4.2 Projection of points on first, second, third and fourth quadrants
- 4.3 Projection of lines: Parallel to one of the principal planes, inclined to one of the principal plane and parallel to other, Inclined to both principal planes, traces of a line
- 4.4 Projection planes: Perpendicular to both principal planes, parallel to one of the principal planes and Inclined to one of the principal planes, perpendicular to other and Inclined to both principal planes
- 4.5 True length of lines: Horizontal, inclined and oblique lines
- 4.6 Rules for parallel and perpendicular lines
- 4.7 Point view or end view of a line
- 4.8 Shortest distance from a point to a line
- 4.9 Edge view and true shape of an oblique plane
- 4.10 Angle between two intersecting lines
- 4.11 Intersection of a line and a plane, visible portion of line
- 4.12 Angle between a line and a plane
- 4.13 Dihedral angle between two planes
- 4.14 Shortest distance between two skew lines
- 4.15 Angle between two non-intersecting (Skew) lines

5 Multi View (Orthographic) Projections

(8 hours)

- 5.1 Orthographic projections
 - 5.1.1 First and third angle projection
 - 5.1.2 Principal views: Methods for obtaining orthographic views, projection of lines, angles and plane surfaces, analysis in three views, projection of curved lines and surfaces, object orientation and selection of views for best representation, full and hidden lines
 - 5.1.3 Orthographic drawings: Making an orthographic drawing, visualizing objects (Pictorial view) from the given views
 - 5.1.4 Interpretation of adjacent areas, true-length lines, representation of holes, conventional practices
- 5.2 Sectional views: Full, half, offset, broken (Partial), rotated/aligned, revolved, removed (Detail) sections, phantom of hidden section, specifying cutting planes for sections, convention practices
- 5.3 Auxiliary views: Basic concept and use, drawing methods and types, symmetrical and unilateral auxiliary views, auxiliary sectional views

6 Developments and Intersections

(7 hours)

- 6.1 Introduction and projection of solids with points transfer
- 6.2 Developments: General concepts and practical considerations, triangulation method for approximate development of surfaces of a right/oblique; Prism, cylinder, pyramid, cone, prism and cylinder cut by oblique planes,

frustum/truncated pyramid and cone, transition pieces for connecting different shapes and sphere

- 6.3 Intersections: Lines of intersection of geometric surfaces, piercing point of a line and a geometric solid, intersection lines of two planes, intersections of – prism and prism, cylinder and prism, cylinder and cylinder, pyramid and prism, cone and prism, pyramid and cylinder, cone and cylinder

7 Pictorial Drawings

(7 hours)

- 7.1 Classifications: Advantages and disadvantages
- 7.2 Isometric view
 - 7.2.1 Axonometric projection
 - 7.2.2 Isometric projection and isometric drawing (View)
 - 7.2.3 Isometric and non-isometric lines; Isometric and non-isometric surfaces
 - 7.2.4 Angles in isometric drawing
 - 7.2.5 Circles and circular arcs in isometric and non-isometric surfaces (Slopes)
 - 7.2.6 Irregular curves in isometric drawing
 - 7.2.7 Isometric sectional views
- 7.3 Oblique drawing
 - 7.3.1 Procedure for making an oblique drawing
 - 7.3.2 Rules for placing objects in oblique drawing
 - 7.3.3 Angles, circles and circular arcs in oblique drawing
- 7.4 Perspective projection
 - 7.4.1 Terms used in perspective projection
 - 7.4.2 Parallel and angular perspective
 - 7.4.3 Selection of station point
 - 7.4.4 Perspective projection of right prism and pyramid solid

Assignments

- 1. Geometrical construction
- 2. Descriptive geometry
- 3. Multi-view Projection I
- 4. Multi-view Projection II
- 5. Surface development and intersection
- 6. Isometric drawing
- 7. Oblique drawing and perspective projection

Practical

(60 hours)

- 1. Drawing sheet layout, freehand lettering, scale, common graphical symbols, sketching of parallel lines, circles, dimensioning
- 2. Geometrical construction (Sketch and instrumental drawing)
- 3. Descriptive geometry I (Sketch and instrumental drawing)
- 4. Descriptive geometry II (Sketch and instrumental drawing)
- 5. Multiview drawings I (Sketch and instrumental drawing)

6. Multiview drawings II (Sketch and instrumental drawing)
7. Multiview, sectional drawings and dimensioning (Sketch and instrumental drawing)
8. Auxiliary view, sectional drawings and dimensioning (Sketch and instrumental drawing)
9. Projection of regular geometrical solids with point transfer (Sketch and instrumental drawing)
10. Surface development of solids I (Sketch and instrumental drawing)
11. Surface development of solids II (Sketch and instrumental drawing)
12. Intersection of solids (Sketch and instrumental drawing)
13. Isometric drawing I (Sketch and instrumental drawing)
14. Isometric drawing II (Sketch and instrumental drawing)
15. Oblique drawing and perspective projection (Sketch and instrumental drawing)

Reference

1. Luzadder, W.J., Duff, J.M. (1989). Fundamentals of engineering drawing: With an introduction to interactive computer graphics for design and production (Latest Edition). Prentice-Hall.
2. French, T.E., Vierck, C.J., Foster, R.J. (1993). Engineering drawing and graphic technology (Latest Edition). McGraw-Hill.
3. Giesecke, F.E., Mitchell, A., Spencer, H.C., Dygdon, J.T. (2016). Technical drawing with engineering graphics. Peachpit Press.
4. Bhatt, N.D. (2023). Elementary engineering drawing. Charotar Publishing House.
5. Gill, P.S. (2016). Engineering drawing. S. K. Kataria & Sons.
6. Dhawan, R.K. (2019). A textbook of engineering drawing. S. Chand and Company.
7. Luintel, M. C. (2021). Engineering Drawing I. Heritage Publishers & Distributors.
8. Luintel, M. C. (2020). Engineering Drawing II. Heritage Publishers & Distributors.

FUNDAMENTAL OF ELECTRICAL AND ELECTRONICS ENGINEERING

ENEX 101

Lecture : 3
Tutorial : 1
Practical : 3

Year : I
Part : I

Course Objectives:

Objective of the course is to understand the language of electronics, elements, and their functionality, to introduce the DC and AC circuit analysis and basic understanding of analog systems and their applications.

1 Basic Circuits Concepts (6 hours)

- 1.1 Current and potential
- 1.2 Passive components: Resistance, inductance, capacitance; Series, parallel combinations; Kirchhoff's voltage and current laws for DC circuits
- 1.3 Signal sources: Voltage and current sources; Non ideal sources; Representation under assumption of linearity; Controlled sources: VCVS, CCCVS, VCCS, CCCS; Concept of gain, transconductance, transimpedance
- 1.4 Maximum power transfer, superposition theorem, Thevenin's theorem, Norton's theorem

2 Average and RMS Values (4 hours)

- 2.1 Generation of AC voltage
- 2.2 Waveform and its characteristics
- 2.3 RMS and average values of periodic waveforms

3 AC Circuit Analysis (12 hours)

- 3.1 Single-phase AC circuit analysis
 - 3.1.1 Series, parallel and network circuits with sinusoidal excitations
 - 3.1.2 The concept of complex impedance and admittance
 - 3.1.3 Sinusoidal excitation of inductive and capacitive reactance and complex impedance
 - 3.1.4 Concept of time phase differences between various sinusoidal quantities
 - 3.1.5 Phasor concept and phasor representation of AC quantities
 - 3.1.6 Transformed impedances and network reduction
 - 3.1.7 Real, reactive and apparent power concepts

- 3.2 Three phase AC circuit
 - 3.2.1 Generation of three phase voltage
 - 3.2.2 Wye and delta connection

4 Diodes (7 hours)

- 4.1 Semiconductor diode characteristics
- 4.2 Modeling the semiconductor diode
- 4.3 Diode circuits: Clipper; Clamper circuits
- 4.4 Zener diode, LED, photodiode, Varactor diode, tunnel diodes
- 4.5 DC power supply: Rectifier-half wave, full wave (Center tapped, bridge), Zener regulated power supply

5 Transistor (10 hours)

- 5.1 BJT configuration and biasing, small and large signal model
- 5.2 T and π model
- 5.3 Concept of differential amplifier using BJT
- 5.4 BJT as switch and logic circuits
- 5.5 Construction and working principle of JFET, MOSFET and CMOS
- 5.6 MOSFET as logic circuits

6 Operational Amplifier and Oscillator (6 hours)

- 6.1 Basic model; Virtual ground concept; Inverting amplifier; Non-inverting amplifier; Integrator; Differentiator, summing amplifier and their applications
- 6.2 Basic feedback theory; Positive and negative feedback; Concept of stability; Oscillator
- 6.3 Waveform generator using op-amp for square wave, triangular wave, phase shift oscillator and Wien bridge oscillator for sinusoidal waveform

Practical (45 hours)

- 1. Familiarization with passive components, function generator and oscilloscope
- 2. Measurement of amplitude, frequency, time period using oscilloscope
- 3. Ohm's law, series, parallel circuits and calculate average, RMS value
- 4. Verification of KCL, KVL and network theorems
- 5. Maximum power transfer/ capacitor charging and discharging
- 6. Diode characteristics, rectifiers, Zener diodes
- 7. BJT characteristics and single stage amplifier
- 8. BJT, PMOS, NMOS and CMOS as switch
- 9. Inverting, non-inverting, summing and subtractor amplifier using Op-amp
- 10. Relaxation oscillator
- 11. Analog sensor and small projects

Reference

- 1. Boylestad, R. L., Nashelsky, L. (2008). Electronic devices and circuit theory (10th ed.). Pearson Education.

2. Floyd, T.L. (2011). Electronic devices (9th ed.). Pearson Education.
3. Sedra, A.S., Smith, K.C. (2014). Microelectronic circuits (7th ed.). Oxford University Press.
4. Cogdell, J.R. (1996). Foundations of electrical engineering (Latest Edition). Prentice Hall.

ENGINEERING PHYSICS

ENSH 102

Lecture : 4
Tutorial : 1
Practical : 2

Year : I
Part : I

Course Objectives:

To provide students a concept and sound knowledge of physics with the emphasis in present day applications to apply them in relevant fields. The background of physics corresponding to proficiency certificate level is assumed.

1 Oscillation (6 hours)

- 1.1 Physical pendulum
 - 1.1.1 Bar pendulum
 - 1.1.2 Interchangeability of point of suspension and point of oscillation
 - 1.1.3 Minimum time period in case of physical pendulum
 - 1.1.4 Torsion pendulum
- 1.2 Damped and forced oscillation
 - 1.2.1 Damped harmonic oscillator
 - 1.2.2 Difference between free and damped oscillator
 - 1.2.3 Energy in damped oscillation
 - 1.2.4 Relaxation time
 - 1.2.5 Forced oscillation and resonance
 - 1.2.6 Sharpness of resonance
 - 1.2.7 Quality factor

2 Acoustics (3 hours)

- 2.1 Introduction
 - 2.1.1 Threshold of hearing and loudness
 - 2.1.2 Reverberation and reverberation time
 - 2.1.3 Absorption coefficient
 - 2.1.4 Sabine's law
 - 2.1.5 Conditions for good acoustics
- 2.2 Ultrasound
 - 2.2.1 Production (Piezoelectric) of ultrasound and its applications
 - 2.2.2 Test of structure and materials
 - 2.2.3 Medical uses

3 Heat and Thermodynamics

(8 hours)

- 3.1 Quantity of heat
 - 3.1.1 Calorific value of foods and fuels
 - 3.1.2 Bomb calorimeter
 - 3.1.3 Specific heat of solid: Dulong-Petit law, Einstein's law
- 3.2 Nature of heat
 - 3.2.1 Degree of freedom
 - 3.2.2 Maxwell's law of equipartition of energy
 - 3.2.3 Atomicity of gases
 - 3.2.4 Vander-Waal's equation of real gases
 - 3.2.5 Critical constants
- 3.3 Thermodynamics
 - 3.3.1 Laws of thermodynamics
 - 3.3.2 Clapeyron latent heat equation
 - 3.3.3 Entropy and Third law of thermodynamics
 - 3.3.4 Negative energy
 - 3.3.5 Maxwell's thermodynamic relations
 - 3.3.6 Gibb's free energy and phase transitions
- 3.4 Heat and mass transfer
 - 3.4.1 Fourier's law of thermal conductivity
 - 3.4.2 Use of thermal conductivity in building sciences
 - 3.4.3 Thermal resistance
 - 3.4.4 Types of convection
 - 3.4.5 Law of diffusion
 - 3.4.6 Relation between Stefan's law and Newton's law of Cooling
 - 3.4.7 Pyrheliometer and Pyrometer

4 Optics

(17 hours)

- 4.1 Geometrical optics
 - 4.1.1 Lens separation
 - 4.1.2 Chromatism in lens combination
- 4.2 Interference
 - 4.2.1 Interference in thin films (Reflected and transmitted light)
 - 4.2.2 Fringes produced by a wedge-shaped thin film
 - 4.2.3 Newton's rings (Both reflected and transmitted case)
 - 4.2.4 Determination of wavelength of light and refractive index of liquid by using Newton's rings.
- 4.3 Diffraction
 - 4.3.1 Introduction: Fresnel and Fraunhofer's diffraction
 - 4.3.2 Fraunhofer's diffraction at single slit
 - 4.3.3 Intensity distribution in the diffraction pattern due to a single slit
 - 4.3.4 Multiple slits, diffraction grating
 - 4.3.5 X-ray diffraction, X-rays in material testing

- 4.4 Polarization
 - 4.4.1 Introduction: double refraction, Nichol prism (Construction and uses)
 - 4.4.2 Retardation plate (Quarter and half wave plates), plane, elliptical and circular polarized light (Theoretical and mathematical explanation)
 - 4.4.3 Optical activity, specific rotation
- 4.5 Laser
 - 4.5.1 Introduction: Laser and ordinary light, properties of laser
 - 4.5.2 Induced absorption, spontaneous and stimulated emission, active medium, population inversion, metastable state
 - 4.5.3 Pumping (Types: Optical, electrical, chemical and thermal)
 - 4.5.4 He-Ne laser, semiconductor laser
 - 4.5.5 Uses of laser
- 4.6 Fiber optics
 - 4.6.1 Introduction: Propagation of light wave
 - 4.6.2 Types of optical fiber: Step index and graded index
 - 4.6.3 Fiber transmission- Single and multimode, self-focusing, acceptance angle and numerical aperture
 - 4.6.4 Applications

5 Electrostatics

(8 hours)

- 5.1 Electric field
 - 5.1.1 Electric field due to an electric dipole (Along axial line and equatorial line)
 - 5.1.2 Electric dipole in an external electric field
 - 5.1.3 Electric field due to linear electric quadrupole (Along axial line)
 - 5.1.4 Electric field: A ring of charge, circular ring and disc of charge
- 5.2 Electric potential
 - 5.2.1 Potential due to electric dipole
 - 5.2.2 Potential due to linear quadrupole
 - 5.2.3 potential due to continuous charge distribution, potential due to ring of charge and disc of charge
- 5.3 Capacitors
 - 5.3.1 Cylindrical capacitor
 - 5.3.2 Charging and discharging of capacitor
 - 5.3.3 Capacitor with dielectrics: dielectrics and Gauss law
 - 5.3.4 High intensity electrostatic fields: Uses and hazards (Xerography, inkjet, precipitation)

6 Electromagnetism

(6 hours)

- 6.1 Electromagnetic induction
 - 6.1.1 Faraday's laws
 - 6.1.2 Induction and energy transformation
 - 6.1.3 Induced electric field
 - 6.1.4 Self-induction and mutual induction
 - 6.1.5 LR circuit
 - 6.1.6 Energy stored in a magnetic field and energy density
 - 6.1.7 Induced magnetic field: Modified Ampere's law and displacement current
- 6.2 Eddy current
 - 6.2.1 Introduction
 - 6.2.2 Applications: Induction cooker, electric guitar, metal detector and Eddy current breaking
 - 6.2.3 Cyclotron and Synchrotron

7 Electromagnetic Waves

(6 hours)

- 7.1 Maxwell's equations
 - 7.1.1 Differential and integral forms
 - 7.1.2 Conversion of Maxwell's equations from integral form to differential form and differential form to integral form
 - 7.1.3 Maxwell's equations in different media
- 7.2 Applications
 - 7.2.1 Wave equations: Non conducting and conducting medium and free space
 - 7.2.2 Plane solution of wave equations, amplitude of electromagnetic waves, speed of electromagnetic waves, ratio of electric and magnetic fields
 - 7.2.3 Continuity equation
 - 7.2.4 Energy transfer and Poynting vector, radiation pressure

8 Photon and Matter Waves

(6 hours)

- 8.1 Quantum physics
 - 8.1.1 Inadequacy of classical mechanics and rise of quantum mechanics, quantization of energy
 - 8.1.2 Group velocity and phase velocity, electrons and matter waves
 - 8.1.3 de-Broglie wavelength, its applications
 - 8.1.4 Heisenberg uncertainty principle and its applications
 - 8.1.5 Wave functions and its significance

- 8.2 Schrodinger wave equation
 - 8.2.1 Time dependent and independent equation
 - 8.2.2 Probability distribution
 - 8.2.3 One dimensional infinite potential well, particle in a box
 - 8.2.4 Barrier tunneling (Reflection and transmission coefficient)

Tutorial

(15 hours)

Students will practice numerical problems covering all units and sub-units to strengthen conceptual understanding and problem-solving skills.

Practical

(30 hours)

1. To determine the acceleration due to gravity and radius of gyration of the given metal bar using bar pendulum
2. To determine the modulus of elasticity of the given material and moment of inertia of the circular disc about the wire as an axis passing through its center and perpendicular to its plane by using torsional pendulum
3. To determine the coefficient of thermal conductivity of a bad conductor by Lee's method
4. To determine the mechanical equivalent of heat by given method
5. To determine the wavelength of the sodium light using Newton's rings
6. To determine the wavelength of sodium light using wedge-shaped method
7. To determine the wavelength of LASER light using diffraction grating and hence determine the particle size of lycopodium powder
8. To determine the focal length of two lenses when they are separated by some finite distance
9. To determine the chromatic aberration of a convex lens between red and blue colors
10. To determine the capacitance of the given capacitor by the method of charging and discharging through resistor
11. To plot the graph between frequency and current in LCR series circuit and hence determine the quality factor of the circuit
12. To study the growth and decay of current in LR circuit then determine the self-inductance of the given inductor
13. To determine the dielectric constant of the given material

Reference

1. Halliday, D., Resnick, R., Walker, J. (2021). Fundamentals of physics. John Wiley & Sons.
2. Pokharel, B., Bhattarai, B.K., Paudel, M.D. (2023). Fundamentals of engineering physics. Benchmark Education Support.
3. Brij Lal, Subrahmanyam, N. (2012). A text book of optics. S. Chand Publishing.
4. Vasudeva, A.S. (2010). Modern engineering physics. S. Chand Publishing.

5. Caur, R.K., Gupta, S.L. (2012). Engineering physics. Dhanpat Rai Publications.
6. Brij Lal, Subrahmanyam, N. (2011). Waves and oscillations. Vikas Publishing House Pvt. Ltd.
7. Brij Lal, Subrahmanyam, N. (2012). Heat, thermodynamics and statistical physics. S. Chand Publishing.
8. Avadhanulu, M.N., Kshirsagar, P.G., Murthy, T.V.S.A. (2018). A textbook of engineering physics. S. Chand Publishing.

ENGINEERING WORKSHOP

ENME 106

Lecture : 1
Tutorial : 0
Practical : 3

Year : I

Part : I

Course Objectives:

After completing this course, the students will be able to practice workshop safety rules effectively with different hand tools and machine tools for producing metal and sheet metal components. Acquire knowledge and practice on casting, forging, welding, soldering, brazing and riveting.

1 Safety Measures in the Workshop (1 hour)

- 1.1 Causes of accident
- 1.2 Types of safety: General safety, personal safety, machine and equipment safety, job safety

2 Bench Work and Fittings (4 hours)

- 2.1 Fitting Tools: Types, uses of holding tools, sticking tools, cutting tools (Files, chisels, hacksaw), scrapping tools (Scrappers), drilling tools (Drill bits), measuring, marking and testing tools (Steel rule, calipers, divider, surface plate, scribe, surface gauge, punches, angle plate, try square, combination sets, vernier caliper, micrometer, bevel protractor, miscellaneous tools (Wrenches, screw drivers and pliers)
- 2.2 Benchwork and fitting operation
- 2.3 Filling operations, chipping operations and sawing operation

3 Thread Cutting (1 hour)

- 3.1 Classification of threads
- 3.2 Thread cutting tools for hand threading
- 3.3 Threading taps: Types, uses and care
- 3.4 Threading dies: Types, uses and care
- 3.5 Thread cutting by hand: Cutting internal and external thread

4 Sheet Metal (2 hours)

- 4.1 Introduction, sheet metal tools, sheet metal operation
- 4.2 Rivet types, types of rivet joints, riveting tools and their uses, riveting procedure

5 Machine Tools (2 hours)

- 5.1 Lathes: Working principle, types of lathes, main parts of lathe, lathe operations (Facing, centre drilling, turning, knurling, boring, chamfering, thread cutting, counter sinking, counter boring)
- 5.2 Drilling Machine: Types of drilling machine, types of drill bits, drilling operations (Drilling, counter boring, reaming, tapping)

6 Forging and Casting (1.5 hours)

- 6.1 Introduction, forging tools, forging operations
- 6.2 Introduction, pattern making foundry tools, core making, sand casting process

7 Welding (2.5 hours)

- 7.1 Arc welding: Introduction, arc welding equipment and accessories, influencing factor in arc welding, methods of striking an arc (Tap, scratch), electrodes, types of joint, welding positions, TIG, MIG welding
- 7.2 Gas welding: Oxyacetylene gas welding, oxyacetylene gas welding accessories, filler rods, fluxes, types of flames and uses

8 Brazing and Soldering (1 hour)

- 8.1 Introduction, brazing equipment and materials, brazing process, surface clearing, joint design, support parts, brazing operations (Heating, filler metal applications, flux application, clearing after brazing)
- 8.2 Introduction, flux, soft solder and soldering process

Practical (45 hours)

- 1. Fitting Practice: Demonstration, usage of different types of hand tools and measuring instruments
- 2. Perform filing, sawing, drilling and tapping operations on given mild steel strip
- 3. Machining practice: Perform lathe operations
- 4. Welding Practice: Perform arc welding and oxyacetylene gas welding operations
- 5. Sheet metal practice: Perform sheet metal operations
- 6. Soldering and brazing
- 7. Electrical installations

Reference

- 1. Khurmi, R.S., Gupta, J.K. (2008). A Textbook of Workshop Technology. S. Chand Publishing.
- 2. Raghuvanshi, B.S. (1990). A Course in Workshop Technology, Volume II (machine tools). New Delhi: Dhanpat Rai & Company.

3. Choudhary, S.K.H., Choudhary, A.K.H. (2005). Elements of Workshop Technology Vol. I and II: Manufacturing Processes. Media promoter & publishers.
4. Khurmi, R. S., Gupta, J. K. (2008). A Textbook of Workshop Technology. S. Chand Publishing.
5. Rajput, R.K. (2007). A textbook of manufacturing technology: Manufacturing processes. Firewall Media.
6. Rao, K.V. (2002). Manufacturing Science and Technology-Manufacturing Processes and Machine Tools. New Age International.
7. Heinrich, G. (2006). All About Machine Tools. New Age International.



YEAR I, PART II

Semester 2

6 COURSES IN THIS SEMESTER

ENSH 151	Engineering Mathematics II	29
ENCT 151	Object Oriented Programming	32
ENEX 152	Digital Logic	36
ENEX 151	Electronic Device and Circuits	40
ENSH 153	Engineering Chemistry	43
ENEE 154	Electrical Circuits and Machines	48

2

ENGINEERING MATHEMATICS II

ENSH 151

Lecture : 3
Tutorial : 2
Practical : 0

Year : I

Part : II

Course Objectives:

After completion of the course students will be able to apply knowledge of partial differentiation, multiple integrals, vector calculus, optimization, matrices and infinite series in their corresponding study area.

1 Calculus of Two and More Variables (6 hours)

- 1.1 Partial differentiation
 - 1.1.1 Partial derivatives of first and higher order
 - 1.1.2 Homogeneous function: Euler's theorem for two and three variables
 - 1.1.3 Total derivatives and differentials, differentiation of composite and implicit functions
 - 1.1.4 Jacobians and their properties
- 1.2 Extreme values of two and three variables. Lagrange's multiplier
- 1.3 Application in optimization of function of two variables in one constraint

2 Multiple Integrals (7 hours)

- 2.1 Double integrals in Cartesian and Polar form, change of order of integration
- 2.2 Triple integrals in Cartesian, cylindrical and spherical coordinates
- 2.3 Area, volume, moment of inertia, mass and centroid by double and triple integrals

3 Vector Calculus (12 hours)

- 3.1 Review of scalar and vector products, scalar and vector triple product, scalar and vector product of four vectors
- 3.2 Vector differentiation and integration, their geometrical meaning, velocity and acceleration
- 3.3 Vector differential operators: Gradient, directional derivatives, divergence and curl
- 3.4 Line integrals, independent of path, conservative and irrotational vector fields, scalar potential
- 3.5 Introduction to Green's theorem and its application
- 3.6 Surface integrals, calculation of flux

- 3.7 Volume integrals, Gauss divergence theorem (Without proof) and its application in evaluation of surface integrals
- 3.8 Introduction to Stoke's theorem and its application

4 Laplace Transform (7 hours)

- 4.1 Definition of Laplace transform, condition for existence, Laplace transforms of some elementary functions, properties of Laplace transform, shifting and change of scale properties
- 4.2 Inverse Laplace transform, uniqueness of inverse Laplace transform, properties of inverse Laplace transform
- 4.3 Laplace transform of derivatives and integral, multiplication and division by t^n the convolution theorem
- 4.4 Laplace transform of Heaviside's unit function, Dirac-delta function and periodic functions
- 4.5 Application of Laplace transform to ordinary differential equations

5 Matrices (8 hours)

- 5.1 Review of algebra of real and complex matrices
- 5.2 Rank of matrices and its application in system of linear equations
- 5.3 Vector space, linear dependence and independence
- 5.4 Eigen values: Cayley Hamilton theorem and its applications
- 5.5 Eigen vectors, diagonalization of matrices
- 5.6 Reduction of quadratic forms into canonical forms (Three variables only)

6 Solution of Differential Equation in Series and Special Functions (5 hours)

- 6.1 Power series method
- 6.2 Bessel's functions: Introduction, properties and application
- 6.3 Legendre's function: Introduction, properties and application

Tutorial (30 hours)

- 1. Techniques of partial differentiation, differentiation of composite and implicit functions, total derivatives, and related exercises
- 2. Exercises related to Euler's theorem
- 3. Exercises related to extreme values of two and three variables
- 4. Change of order of integration in multiple integrals
- 5. Exercises related to application of double and triple integrals in finding area, volume, moment of inertia, mass and centroid
- 6. Examples related to revision of scalar and vector product of two and three vectors
- 7. Problems on gradient, directional derivatives, divergence and curl
- 8. Exercises on line integrals, independent of path
- 9. Exercises on surface integrals
- 10. Exercises on Green's theorem, verification and application in calculating line integrals
- 11. Verification of Stoke's theorem, application

12. Verification of Gauss' Divergence theorem, and application in calculating surface integrals
13. Exercises related to Laplace transforms
14. Exercises on inverse Laplace transforms
15. Problems related to application of Laplace transform to ordinary differential equations
16. Examples related to Laplace transform of Heaviside's unit function, Dirac-delta function and periodic functions
17. Rank and solution of simultaneous equations
18. Eigen values and Eigen vectors, diagonalization problems
19. Problems related to reduction of quadratic forms into canonical forms
20. Exercises related to Bessel's function and Legendre's polynomial

Final Exam

The questions will cover all the chapters in the syllabus. The evaluation scheme will be as indicated in the table below:

Chapter	Hours	Mark distribution*
1	6	8
2	7	8
3	12	18
4	7	8
5	8	12
6	5	6
Total	45	60

* There may be minor deviation in marks distribution.

References

1. Kreyszig, E. (2011). Advanced engineering mathematics. John Wiley & Sons.
2. Jeffrey, A. (2002). Advanced engineering mathematics. Academic Press.
3. O'Neil, P.V. (2011). Advanced engineering mathematics. Cengage Learning.
4. Sastry, S.S. (2008). Engineering mathematics (Vols. I–II). PHI Learning.
5. Wylie, C.R, Barrett, L.C. (1995). Advanced engineering mathematics (Latest Edition). McGraw-Hill.
6. Dutta, D. (2006). Textbook of engineering mathematics (Vols. I–II). New Age International.

OBJECT ORIENTED PROGRAMMING

ENCT 151

Lecture : 3
Tutorial : 1
Practical : 3

Year : I
Part : II

Course Objectives:

To provide students with foundation in understanding and applying principles of object oriented programming. Emphasis will be given on developing object oriented programming skills using C ++.

1 Introduction to Object Oriented Programming (3 hours)

- 1.1 History of programming languages
- 1.2 Need of object oriented programming(OOP)
- 1.3 Object oriented programming versus procedure oriented programming
- 1.4 Concepts of object oriented programming
- 1.5 Popular object oriented languages
- 1.6 Advantages of OOP
- 1.7 Disadvantages of OOP

2 Basics of C++ Programming (5 hours)

- 2.1 C++ Program structure
- 2.2 Character set and tokens
- 2.3 Variable declaration and expression
- 2.4 Data types
- 2.5 Type conversion and casting
- 2.6 User defined constant const
- 2.7 Reference variables
- 2.8 Conditions and looping
- 2.9 Namespace scope
- 2.10 Functions
 - 2.10.1 Function overloading
 - 2.10.2 Inline function
 - 2.10.3 Default arguments
 - 2.10.4 Pass by reference and return by reference
- 2.11 Array, pointer and string
- 2.12 Structure and unions
- 2.13 Enumeration
- 2.14 Dynamic memory allocation

3 Objects and Classes

(7 hours)

- 3.1 C++ classes
- 3.2 Objects and the member access
- 3.3 Relation of object, class and memory
- 3.4 Defining member function
- 3.5 Defining outer function inline
- 3.6 Objects as member
- 3.7 Constructors and destructors
- 3.8 Object as function arguments
- 3.9 Returning objects from functions
- 3.10 Array of objects
- 3.11 Pointer to objects
- 3.12 Dynamic memory allocation for objects
- 3.13 Dynamic constructors
- 3.14 Pointer
- 3.15 Static data member and static member function
- 3.16 Constant member functions and constant objects
- 3.17 Friend functions and friend classes

4 Operator Overloading

(5 hours)

- 4.1 Overloadable and non- overloadable operators
- 4.2 Syntax of operator overloading
- 4.3 Operator overloading using member operator functions
- 4.4 Operator overloading using non member functions
- 4.5 Unary operator overloading
- 4.6 Binary operator overloading
- 4.7 Type conversion between objects
- 4.8 Explicit constructors

5 Inheritance

(5 hours)

- 5.1 Base class and derived class
- 5.2 protected access specifier
- 5.3 Derived class declaration
- 5.4 Is_a relation and Has_a relation
- 5.5 Public, protected and private inheritance
- 5.6 Member overriding
- 5.7 Forms of inheritance
- 5.8 Constructors in derived class
- 5.9 Destructor in derived class
- 5.10 Need of virtual base class

6 Virtual Functions

(4 hours)

- 6.1 What is virtual function
- 6.2 Need of virtual function
- 6.3 Pointer to derived class
- 6.4 Array of pointers to base class
- 6.5 Pure virtual functions and abstract class
- 6.6 Virtual destructors
- 6.7 Reinterpret cast operator
- 6.8 Run-time type information

7 Stream Computation

(6 hours)

- 7.1 Input/output stream class hierarchy
- 7.2 Testing stream errors
- 7.3 Unformatted input/output and formatted input/output
- 7.4 Stream operator overloading
- 7.5 File input/output with streams
- 7.6 File stream class hierarchy
- 7.7 ASCII and binary files
- 7.8 Read/Write from file
- 7.9 Sequential access to file
- 7.10 Random access to file
- 7.11 File access pointers and their manipulators
- 7.12 Testing errors during file operations
- 7.13 File input/output with member functions

8 Templates

(6 hours)

- 8.1 Function template
- 8.2 Overloading function template
- 8.3 Class template
- 8.4 Derived class template
- 8.5 Introduction to standard template library
 - 8.5.1 Components of STL
 - 8.5.2 Container
 - 8.5.3 Iterators
 - 8.5.4 Algorithms

9 Exception Handling

(4 hours)

- 9.1 Basics of exception handling
- 9.2 Advantage over conventional error handling
- 9.3 Exception handling mechanism
- 9.4 Multiple handlers
- 9.5 Catching all exceptions

- 9.6 Rethrowing exception
- 9.7 Exception with arguments
- 9.8 Exceptions specification for function
- 9.9 Exceptions in constructors and destructors
- 9.10 Handling uncaught exceptions
- 9.11 Handling unexpected exception

Tutorial (15 hours)

After completing each chapter some problems are solved and students are asked to solve programming problems with the teacher's assistance.

Assignment

Appropriate assignment problems are given to students after the completion of each chapter.

Practical (45 hours)

There will be nine laboratory exercises encompassing the entire course content. At the end of the course, students are required to submit a programming project demonstrating the application of object-oriented programming concepts.

Final Exam

The questions will cover all the chapters in the syllabus. The evaluation scheme will be as possible as indicated in the table below:

Chapter	Hours	Marks Distribution*
1	3	4
2	5	7
3	7	9
4	5	7
5	5	7
6	4	5
7	6	8
8	6	8
9	4	5
Total	45	60

* There may be minor deviation in marks distribution.

References

1. Schildt, H. (2015). C++: The Complete Reference. McGraw-Hill Education.
2. Baral, D.S., Baral, D. (2010). Secrets of Object Oriented Programming in C++. Bhundipuran Prakasan.
3. Lafore, R. (2002). Object Oriented Programming in C++. Sams Publishing.
4. Deitel, H.M., Deitel, P.J. (2007). C++ How to Program. Prentice Hall.

DIGITAL LOGIC

ENEX 152

Lecture : 3
Tutorial : 1
Practical : 3

Year : I
Part : II

Course Objectives:

This course mainly focuses on study, analyze basic principle, design and applications of digital circuitries in various fields. It also shows an important branch of the electronics that revolutionizes the modern digital world.

1 Introduction (5 hours)

- 1.1 Digital versus analog signals
- 1.2 Logic level diagram
- 1.3 Digital integrated circuits (ICs)
- 1.4 Clock triggering systems
- 1.5 Digital system applications
- 1.6 Digital codes and conversions
 - 1.6.1 Decimal, binary, octal and hexadecimal codes
 - 1.6.2 BCD code
 - 1.6.3 Excess-3 code
 - 1.6.4 Gray code
 - 1.6.5 Examples of code conversions
- 1.7 Alphanumeric codes: ASCII code and EBCDIC code
- 1.8 1's complement and 2's complement
- 1.9 Signed number representation

2 Logic Gates (3 hours)

- 2.1 Basic gates and their equivalents
- 2.2 Universal gates and their equivalents
- 2.3 Exclusive gates and their equivalents
- 2.4 Positive and negative logic
- 2.5 De'Morgan's laws
- 2.6 Applications of logic gates

3 Boolean Algebra and K-Maps (4 hours)

- 3.1 Boolean algebra and its laws
- 3.2 Simplifications of Boolean expressions
- 3.3 Minterms and maxterms
- 3.4 Sum-of-product and product-of-sum methods

- 3.5 Truth tables and Karnaugh map
- 3.6 Four variables K-maps.
- 3.7 Cell, pairs, quads and octets
- 3.8 Rolling, envelop effects and redundant groups
- 3.9 Don't care conditions

4 Combinational Logic Circuits

(8 hours)

- 4.1 Design procedures
- 4.2 Half-adder and full-adder
- 4.3 Half-subtractor and full-subtractor
- 4.4 Ripple carry adders and fast adders
- 4.5 Multiplexers design
- 4.6 Demultiplexers design
- 4.7 Basic encoders
- 4.8 Priority encoders
- 4.9 Encoder designs
- 4.10 Decoder designs
- 4.11 BCD-to-decimal decoder
- 4.12 Seven-segment decoder
- 4.13 Magnitude comparators

5 Sequential Logic Circuits

(5 hours)

- 5.1 Latches and flip-flops: SR, D, T and JK
- 5.2 Excitation tables, characteristic equations
- 5.3 Master-slave flip-flops
- 5.4 Flip-flop timing diagrams
- 5.5 Flip-flops as the state machines
- 5.6 Flip-flop conversions
- 5.7 Flip-flop applications

6 Registers and Counters

(7 hours)

- 6.1 Register fundamentals, register types
- 6.2 SISO, SIPO, PISO and PIPO registers
- 6.3 Data transfer timing diagrams
- 6.4 Asynchronous counters
- 6.5 Up, down and mod-n asynchronous counters
- 6.6 Synchronous counters
- 6.7 Up, down and mod-n synchronous counters
- 6.8 Register and counter applications

7 Sequential Machine Designs

(8 hours)

- 7.1 Machine design procedures

- 7.2 Primitive state diagrams
- 7.3 Transition/flow tables
- 7.4 Redundant states
- 7.5 Pure binary assignment tables
- 7.6 Excitation maps
- 7.7 Realization of the models
- 7.8 Circuit diagram of synchronous machine
- 7.9 One-bit and two-bit input sequence detectors

8 Digital Integrated Circuits

(5 hours)

- 8.1 BJT and MOSFET switching circuits
- 8.2 TTL parameters
- 8.3 TTL circuits: NAND, NOT, NOR
- 8.4 CMOS parameters
- 8.5 CMOS logic circuits: NAND, NOR, NOT
- 8.6 Three-state TTL devices
- 8.7 Digital devices applications
 - 8.7.1 Multiplexing displays
 - 8.7.2 Frequency counters
 - 8.7.3 Time measurements

Tutorial

(15 hours)

- 1. Different code conversion examples
- 2. Sign numbers addition and subtraction
- 3. Realization of positive and negative logic gates
- 4. Application of Boolean algebra and K-map for various logic designs
- 5. Multiplexer tree concepts
- 6. Realization of adder/subtractor using multiplexers
- 7. Demultiplexer tree concepts
- 8. Realization of adder/subtractor using demultiplexers
- 9. 16 - to – 4 line encoder design
- 10. Octal and decimal priority encoder designs
- 11. 4– to – 16 line decoder design
- 12. BCD-to-decimal decoder design
- 13. Any segment of 7-segment decoder design
- 14. Concept of designing n-bit magnitude comparator
- 15. Flip-flop conversion from one flip-flop to another type
- 16. Shift register timing diagram practice
- 17. Ripple counter design concept
- 18. Decade synchronous counter design
- 19. Up and down counter in a single circuit
- 20. 3-bit and 4-bit binary sequence detector synchronous machine design

Practical

(45 hours)

- 1. Basic gates, universal gates and exclusive gates

2. De' Morgan's law and its familiarization with NAND and NOR Gates
3. Encoders and decoders
4. Multiplexers and demultiplexers
5. Binary addition and subtraction
6. Latches, RS, and T flip-flops.
7. D and JK flip-flop and master-slave flip-flop
8. Shift registers
9. Circuit realizations on ripple counters
10. Circuit realizations on synchronous counters

Final Exam

The questions will cover all the chapters in the syllabus. The evaluation scheme will be as indicated in the table below:

Chapter	Hours	Marks distribution*
1	5	7
2	3	4
3	4	5
4	8	10
5	5	7
6	7	10
7	8	10
8	5	7
Total	45	60

* There may be minor deviation in marks distribution.

References

1. Floyd, T. L. (2015). Digital fundamentals. Pearson Education.
2. Mano, M. M. (1995). Digital design (Latest Edition). Prentice Hall.
3. Leach, D.P., Malvino, A.P., Saha, G. (2012). Digital principles and applications. Tata McGraw-Hill Education.
4. Fletcher, W.I. (1980). An engineering approach to digital design (Latest Edition). Prentice-Hall.
5. Gothmann, W.H. (1982). Digital electronics: An introduction to theory and practice (Latest Edition). Prentice-Hall.

ELECTRONIC DEVICE AND CIRCUITS

ENEX 151

Lecture : 3

Tutorial : 1

Practical : 3

Year : I

Part : II

Course Objectives:

To introduce the fundamentals of analysis of electronic circuits and to provide basic understanding of semiconductor devices and analog integrated circuits.

1 The Bipolar Junction Transistor (BJT) (9 hours)

- 1.1 Review of operation of the npn transistor in the active mode
- 1.2 Review of graphical representation of transistor characteristics
- 1.3 Analysis of transistor circuits at DC
- 1.4 Graphical DC load line analysis
- 1.5 Transistor as an amplifier (r_{π} , r_e , g_m)
- 1.6 Biasing BJT for discrete-circuit design
- 1.7 Small signal equivalent circuit models (π and T)
- 1.8 Basic single-stage BJT amplifier configuration (C-B, C-E, C-C)
- 1.9 Small signal analysis of amplifier
- 1.10 Transistor as a switch – cutoff and saturation
- 1.11 A general large-signal model of the BJT: The Ebers-Moll model

2 Field-Effect Transistor (10 hours)

- 2.1 Structure and physical operation of the junction field-effect transistor
- 2.2 Structure and physical operation of enhancement-type MOSFET
- 2.3 Current-voltage characteristic of enhancement-type MOSFET
- 2.4 The depletion-type MOSFET
- 2.5 Biasing in MOS amplifier circuits
- 2.6 MOSFET circuits at DC
- 2.7 MOSFET as an amplifier (Common source)
- 2.8 MOSFET and CMOS as logic circuits

3 Operational Amplifier Circuits and Oscillator (10 hours)

- 3.1 Review of basic principles of sinusoidal oscillator
- 3.2 Review of Op-Amp square and triangular, RC oscillator circuits
- 3.3 LC and crystal oscillators
- 3.4 Integrated circuit timers
- 3.5 Precision rectifier circuits
- 3.6 Bias circuits suitable for IC design
- 3.7 The Widlar current source

- 3.8 The differential amplifier
- 3.9 Active loads
- 3.10 Output stages

4 Output Stages and Power Amplifiers (10 hours)

- 4.1 Classification of output stages
- 4.2 Class A output stage
- 4.3 Class B output stage
- 4.4 Class AB output stage
- 4.5 Biasing of class AB output stage
- 4.6 Power BJT's
- 4.7 Transformer-coupled push-pull stage
- 4.8 Tuned amplifiers

5 Power Supplies, Breakdown Diodes, and Voltage Reference (6 hours)

- 5.1 Unregulated power supply
- 5.2 Zener regulated power supply
- 5.3 Zener diodes, bandgap voltage reference, constant current diodes
- 5.4 Transistor shunt/series voltage regulator
- 5.5 Improving voltage regulator performance with feedback
- 5.6 IC voltage regulator

Tutorial (15 hours)

The tutorial sessions will focus on chapter-specific exercises aimed at enhancing understanding and application in electronic device and circuits.

Practical (45 hours)

1. Diode characteristics, rectifiers, Zener diodes
2. Bipolar junction transistor characteristics and single stage amplifier
3. BJT single stage amplifier (R_{in} , R_{out} , Gain)
4. Power amplifiers
5. Field effect transistor characteristics
6. FET single stage amplifier
7. BJT differential amplifier
8. Relaxation oscillator and sinusoidal oscillator (Phase shift, Wien bridge)
9. Series, shunt and IC voltage regulators
10. Multivibrator using 555 timer IC
11. Project presentation

Final Exam

The questions will cover all the chapters in the syllabus. The evaluation scheme will be as indicated in the table below:

Chapter	Hours	Mark distribution*
1	9	12
2	10	13
3	10	13
4	10	13
5	6	9
Total	45	60

* There may be minor deviation in marks distribution.

References

1. Sedra, A.S., Smith, K.C. (2010). Microelectronic circuits. Oxford University Press.
2. Boylestad, R.L., Nashelsky, L. (2013). Electronic devices and circuit theory. Pearson.
3. Floyd, T. L. (2013). Electronic devices. Pearson Education Limited.
4. Millman, J., Halkias, C., Parikh, C.D. (2010). Integrated electronics: Analog and digital circuits and systems. Tata McGraw-Hill Education.
5. Bell, D.A. (2009). Electronic devices and circuits. Oxford University Press.

ENGINEERING CHEMISTRY

ENSH 153

Lecture : 3
Tutorial : 1
Practical : 3

Year : I
Part : II

Course Objectives:

To develop the basic concepts of physical chemistry, inorganic chemistry, analytical chemistry, environmental chemistry, green and sustainable chemistry, nano chemistry, polymer chemistry and organic chemistry relevant to the different disciplines of engineering.

1 Electrochemistry and Buffer

(8 hours)

- 1.1 Electrochemistry
 - 1.1.1 Introduction
 - 1.1.2 EMF of galvanic cell, Nernst equation
 - 1.1.3 Polarization and overpotential
 - 1.1.4 Butler-Volmer equation and Tafel plots
- 1.2 Electrode processes and mechanisms (Qualitative only)
 - 1.2.1 Charge transfer processes at electrodes
 - 1.2.2 Mass transfer and diffusion in electrochemical systems
- 1.3 Industrial and applied electrochemistry
 - 1.3.1 Batteries: Lead acid and lithium ion
 - 1.3.2 Solar-photovoltaic cell (With typical examples), fuel cell
 - 1.3.3 Corrosion
- 1.4 Buffer, buffer range, buffer capacity and buffer solution (Henderson-Hasselbalch equation) and its applications

2 Catalyst and Catalysis

(4 hours)

- 2.1 Definition and types
- 2.2 Design and criteria
 - 2.2.1 Structure-activity relationships
 - 2.2.2 Selection criteria of catalyst
- 2.3 Photocatalysis and electrocatalysis
- 2.4 Catalysis for energy and environmental applications
 - 2.4.1 Catalytic conversion of fossil fuels
 - 2.4.2 Renewable energy catalysts
 - 2.4.3 Catalyst for pollution control

- 3 Analytical Techniques and their Applications (6 hours)**
- 3.1 Chromatography
 - 3.2 Mass spectroscopy
 - 3.3 X-ray diffraction (XRD)
 - 3.4 UV-visible spectroscopy
 - 3.5 Infrared-spectroscopy (IR)
 - 3.6 Nuclear magnetic resonance spectroscopy (NMR)
- 4 Metal Complexes, Rare Earth Elements and Metal alloys (6 hours)**
- 4.1 Complexes
 - 4.1.1 Introduction and Werner's theory
 - 4.1.2 Geometry of complex by VBT and its applications
 - 4.1.3 Crystal field theory: Principle and applications
 - 4.2 Rare earth elements: Introduction and applications
 - 4.3 Metallic alloys and applications
- 5 Sustainable Chemistry (7 hours)**
- 5.1 Green chemistry: Introduction and principles
 - 5.2 Water chemistry
 - 5.2.1 Importance of water quality standards
 - 5.2.2 Degree of hardness, scale formation in boiler and softening of hard water
 - 5.2.3 Water pollution with reference to turbidity, COD, BOD, heavy metals, radioactive substances, and plastic
 - 5.2.4 Industrial wastewater and its treatment
 - 5.3 Air pollution: Particulate matter, SO_x, NO_x, GHGs, VOCs, their impacts and remedies
 - 5.4 Waste management
 - 5.4.1 Segregation and management of solid waste
 - 5.4.2 Management of biodegradable waste into energy
 - 5.4.3 E-waste and its management
- 6 Nanoscience and Nanotechnology (3 hours)**
- 6.1 Introduction and types of nano materials (0-, 1-, 2-, and 3- dimensional)
 - 6.2 Nanoparticles, nanofibers, nanowires, carbon nanotubes, graphene, mxene, quantum dots, and their uses
 - 6.3 Preparation of nanomaterials
- 7 Engineering Materials (7 hours)**
- 7.1 Polymers
 - 7.1.1 Natural and synthetic, organic and inorganic, conducting and non-conducting

- 7.1.2 Types of polymerizations: Addition and condensation polymerization
- 7.1.3 Preparation and applications of epoxy resin, polyurethane, Kevlar, polycarbonate, polymethyl methacrylate, polyacrylonitrile, silicones; Phosphorus based polymer, Sulphur based polymer
- 7.1.4 Conducting polymers: Synthesis and application
- 7.1.5 Composite: Fiber reinforced polymer
- 7.1.6 Natural polymers: Cellulose, chitin, chitosan, collagen
- 7.2 Cement: Hydration and setting chemistry of cement

8 Explosives, Lubricants and Paints

(4 hours)

- 8.1 Explosives
 - 8.1.1 Types of explosives: Primary, low and high explosives
 - 8.1.2 Preparation and applications of TNT, TNG, Nitrocellulose and plastic explosives
- 8.2 Lubricants: Introduction, function and classification
- 8.3 Paints
 - 8.3.1 Introduction, requisites, types and applications
 - 8.3.2 Environmental and health impact

Tutorial

(15 hours)

- 1. Introduction to cells, electroplating, EMF, electric double layer, problems related to buffer and Nernst equations
- 2. Types of catalyst and types of catalysis
- 3. Electromagnetic radiation, electromagnetic spectrum, electromagnetic wave, principles of spectroscopy, types of molecular spectra
- 4. Complexes, ligands, postulates, compounds with coordination number 4 and 6, splitting of octahedral and tetrahedral complexes and rare earth elements
- 5. Application of green chemistry and industrial waste management
- 6. Application of nanomaterials in pollution minimization
- 7. Introduction and stabilization of free radical, carbocation and carbanion.
- 8. Exothermic reaction of cement and its applications
- 9. Introduction of and applications of explosives, lubricants and paints

Practical

(45 hours)

- 10. Determination of total, temporary and permanent hardness of water sample using complexometric titration
- 11. Determination of the alkalinity of water sample A and B by double indicator titration
- 12. Estimation of the amount of residual chlorine in water by iodometric titration
- 13. Preparation of the standard buffer solution (Acidic or basic) and measure the approximate pH of given unknown solution by using Universal Indicator

14. Comparison of the cleansing power of two sample of detergents by determining the reduction they cause in surface tension of water
15. Construction of Daniell cell and study of the variation of cell potential with concentration
16. Separation of the pigments through the process of paper / thin layer chromatography
17. Determination of total iron in ground water using spectrophotometer technique
18. Determination of amount of copper and iron in a given mixture solution by $K_2Cr_2O_7$ titration
19. Preparation of cross – linked polymer by condensation polymerization method
20. Standardize potassium permanganate solution and use it to estimate the amount of Iron and determine the percentage purity in the sample of ferrous salt solution
21. Preparation of Ni-DMG complex and estimation of the amount of nickel

Final Exam

The questions will cover all the chapters in the syllabus. The evaluation scheme will be as indicated in the table below:

Chapter	Hours	Marks Distribution*
1	8	10
2	4	5
3	6	5
4	6	10
5	7	10
6	3	5
7	7	10
8	4	5
Total	45	60

*There may be minor deviation in mark distribution

References

1. Maron, S. H., Prutton, C. F. (1965). Principles of Physical Chemistry (Latest Edition). Macmillan.
2. Lee, J.D. (2007). Concise Inorganic Chemistry. John Wiley & Sons.
3. Madan, R.D., Prakash, S. (1994). Inorganic Chemistry (Latest Edition). S. Chand & Company Ltd.
4. Bahl, S., Tuli, G.D., Bahl, A. (2009). Essentials of Physical Chemistry. S. Chand & Co. Ltd.
5. Bhagi, A. K., Morrison, G. R. T., Boyd, R. N. (2008). Organic Chemistry. Prentice-Hall of India.
6. Morrison, R. T., Boyd, R. N. (2008). Organic Chemistry. Prentice-Hall of India.

7. Vogel. (2008). Vogel's Textbook of Quantitative Chemical Analysis. Pearson Education.
8. Murthy, B. S., Shankar, P., Baldev, R., Rath, B. B., Murday, J. (2012). Textbook of nanoscience and nanotechnology. Universities Press.
9. Chatwal, G.R. (2023). Textbook of Environmental Chemistry. Himalaya Publishing House.

ELECTRICAL CIRCUITS AND MACHINES

ENEE 154

Lecture : 4
Tutorial : 1
Practical : 3/2

Year : I
Part : II

Course Objectives:

To develop a comprehensive understanding of electric circuit theory and analysis techniques, alongside the principles and operation of electric machines including transformers, DC and AC machines.

1 Transients in Electric Circuit (7 hours)

- 1.1 Characteristics of various network elements
- 1.2 Nodal analysis with dependent and independent sources
- 1.3 Mesh analysis with dependent and independent sources
- 1.4 Application of matrix method in network analysis
- 1.5 Procedure of evaluating initial conditions
- 1.6 Initial values of derivatives
- 1.7 Initial condition in the case of R-L-C network

2 Transient Analysis R-L-C Circuit by Classical Method (10 hours)

- 2.1 Introduction
- 2.2 First order differential equation with constant coefficient
- 2.3 Higher order homogenous and non-homogenous differential equation with constant coefficient
- 2.4 Particular integral by method of undetermined coefficient
- 2.5 Response of R-L and R-C circuits with DC excitation
 - 2.5.1 DC excitation
 - 2.5.2 Exponential excitation
 - 2.5.3 Sinusoidal excitation
- 2.6 Response of Series R-L-C circuits with
 - 2.6.1 DC excitation
 - 2.6.2 Exponential excitation
 - 2.6.3 Sinusoidal excitation
- 2.7 Response of Parallel R-L-C circuits with
 - 2.7.1 DC excitation
 - 2.7.2 Exponential excitation

- 3 Transient Analysis Using Laplace Transform (7 hours)**
- 3.1 Introduction
 - 3.2 Response of R-L and R-C circuits with
 - 3.2.1 DC excitation
 - 3.2.2 Exponential excitation
 - 3.2.3 Sinusoidal excitation
 - 3.3 Response of series R-L-C circuits with
 - 3.3.1 DC excitation
 - 3.3.2 Exponential excitation
 - 3.3.3 Sinusoidal excitation
 - 3.4 Response of parallel R-L-C circuits with
 - 3.4.1 DC excitation
 - 3.4.2 Exponential excitation
- 4 Network Transfer Function and Frequency Response (8 hours)**
- 4.1 Concept of complex frequency
 - 4.2 Transfer functions of two port networks
 - 4.3 Poles and zeros of networks
 - 4.4 Magnitude and phase response
 - 4.5 Bode diagrams
 - 4.6 Band width, high-q and low-q circuits
 - 4.7 Basic concept of filters: High-pass, low-pass, band-stop and band-pass filters
- 5 Two-Port Parameters of Network (8 hours)**
- 5.1 Definitions of two-port networks
 - 5.2 Parameters of two-port networks
 - 5.2.1 Open circuit impedance parameters
 - 5.2.2 Short circuit admittance parameters
 - 5.2.3 Transmission line parameters
 - 5.2.4 Inverse transmission line parameters
 - 5.2.5 Hybrid parameters
 - 5.2.6 Inverse hybrid parameters
 - 5.3 Relationship and transformation between sets of parameters
 - 5.4 Interconnection of two port networks
 - 5.5 Condition for reciprocity and symmetry
- 6 Magnetic Circuit and Induction (3 hours)**
- 6.1 Magnetic circuit and its types
 - 6.2 B-H relationship and hysteresis with DC excitation
 - 6.3 Hysteresis with AC excitation
 - 6.4 Hysteresis loss and Eddy current loss

- 6.5 Faraday's law of electromagnetic induction, statically and dynamically induced EMF
- 6.6 Force on current carrying conductor

7 Transformers (6 hours)

- 7.1 Construction, operating principle and EMF equation of single-phase transformer
- 7.2 No load and load operation of transformer
- 7.3 Equivalent circuit diagram of transformer
- 7.4 Transformer testing (Open circuit and short circuit)
- 7.5 Voltage regulation, losses, efficiency and condition for maximum efficiency
- 7.6 Auto transformer, Isolation transformer

8 DC Machine (5 hours)

- 8.1 Constructional details of DC machine
- 8.2 Operating principle and EMF equation of DC generator
- 8.3 Operating principle and torque equation of DC motor
- 8.4 Types of DC machine
- 8.5 Back EMF and its role in DC motor
- 8.6 Performance characteristics of DC motor
- 8.7 Starting of DC motor using 3-point starter
- 8.8 Speed control of DC motor (Armature control, field control)
- 8.9 Losses and efficiency

9 AC Motor (6 hours)

- 9.1 Construction, production of rotating magnetic field and operating principle of three-phase induction motor
- 9.2 Torque equation of three-phase induction motor at standstill and running condition
- 9.3 Torque slip characteristics, condition for maximum torque and effect of rotor resistance on torque slip characteristics
- 9.4 Single-phase induction motor
- 9.5 Double field revolving theory
- 9.6 Starting of single-phase induction motor (Capacitor start and run, shaded pole)
- 9.7 Introduction to permanent magnet brushless DC motor, hysteresis motor, stepper motor, servo motor, universal motor

Tutorial (15 hours)

The tutorial sessions will focus on chapter-specific exercises aimed at enhancing understanding and application in electrical circuits and machines.

Practical (22.5 hours)

- 1. Transient response in first order system passive circuit

- Measure step and impulse response RL and RC circuit using oscilloscope
- Relate time response to analytical transfer function calculation
- 2. Transient response in second order system passive circuit
 - Measure step and impulse response RLC series and parallel circuit using oscilloscope
 - Relate time response to analytical transfer function and pole-zero configuration
- 3. Two port network
 - To calculate and verify 'ABCD' parameters of two-port network
 - To determine equivalent parameters of parallel connection of two-port network
- 4. Two winding transformers
 - Measure amplitude and phase response and plot Bode diagram for RLC circuits
 - To perform open circuit (OC) and short circuit (SC) test to determine
- 5. DC motor
 - speed control of DC Shunt motor by (a) armature control method (b) field control method
 - To observe the effect of increasing load on DC shunt motor's speed, armature current and field current
- 6. Single phase AC motors
 - To study the effect of a capacitor on the starting and running of a single-phase induction motor
 - Reversing the direction of rotation of a single-phase capacitor start induction motor

Final Exam

The questions will cover all the chapters in the syllabus. The evaluation scheme will be as indicated in the table below:

Chapter	Hours	Mark distribution*
1	7	7
2	10	10
3	7	7
4	8	8
5	8	8
6	3	3
7	6	6
8	5	5
9	6	6
Total	60	60

* There may be minor deviation in marks distribution.

References

1. Van Valkenburg, M. E. (2019). Network analysis. Pearson Education.
2. Hayt, W.H., Kemmerly, J.E., Phillips, J.D., Durbin, S.M. (2019). Engineering circuit analysis. McGraw-Hill Education.
3. Ciletti, M.D. (1995). Introduction to circuit analysis and design (Latest Edition). Oxford University Press.
4. Soni, K.M. (2013). Circuits and systems. S. K. Kataria & Sons.
5. Nagrath, F.I.J., Kothari, D.P. (2017). Electric machines. McGraw Hill Education.
6. Fitzgerald, A.E., Kingsley, C. (2017). Electric machinery. McGraw Hill Education.
7. Sahdev, S.K. (2018). Electrical machines. Cambridge University Press.
8. Hussain, A. (2016). Electrical machines. Dhanpat Rai & Co.



YEAR II, PART I

Semester 3

6 COURSES IN THIS SEMESTER

ENSH 201	Engineering Mathematics III	54
ENSH 204	Communication English	57
ENCT 201	Computer Graphics and Visualization	60
ENCT 202	Foundation of Data Science	64
ENCT 203	Theory of Computation	68
ENEX 201	Microprocessors	71



ENGINEERING MATHEMATICS III

ENSH 201

Lecture : 3
Tutorial : 2
Practical : 0

Year : II
Part : I

Course Objectives:

The objective of this course is to equip students with understanding and practical application of Fourier series, Fourier transform, function of complex variable, partial differential equations and obtaining mathematical models and Z- transform.

1 Fourier Series and Fourier Transform (12 hours)

- 1.1 Review of periodic, odd and even functions
- 1.2 Fourier series of a function over an interval of length 2ℓ and 2π ; Euler's formula, Dirichlet's condition for uniform convergence of Fourier series, Fourier series of discontinuous functions
- 1.3 Half range Fourier sine and cosine series
- 1.4 Complex form of Fourier series; frequency and amplitude of a function
- 1.5 Fourier integral theorem, Fourier sine and cosine integrals, complex form of Fourier integral
- 1.6 Fourier transform, Fourier sine transform, Fourier cosine transform and their inversion formulas
- 1.7 Fourier transform of the derivative of a function
- 1.8 Relation between Fourier and Laplace transform

2 Functions of Complex Variable (12 hours)

- 2.1 Intuitive idea of limit, continuity and differentiability of functions of complex variable
- 2.2 Analytic functions, the Cauchy Reimann equations both in Cartesian and polar form, construction of analytic functions
- 2.3 Harmonic functions, the orthogonal system
- 2.4 Application of analytic functions in flow problems
- 2.5 Transformation (Mapping), conformal mapping, translation, rotation and magnification; inversion, bilinear transformation
- 2.6 Complex integration, simply and multiply connected regions, Cauchy's integral theorem and formula
- 2.7 Series of complex terms, power series, circle of convergence and radius of convergence, Taylor's and Laurent's series
- 2.8 Zeros, singularities, poles; residue at poles, Cauchy's residue theorem and evaluation real and improper integrals

- 3 Partial Differential Equations (5 hours)**
- 3.1 Definition and formation of partial differential equations
 - 3.2 Partial differential equations solvable by direct integration
 - 3.3 Linear partial differential equation of the first order, Lagrange's linear equations and their solution
 - 3.4 Nonlinear partial differential equation of first order; equations of the form $f(p, q) = 0$, $z = px + qy + f(p, q)$, $f(z, p, q) = 0$, $f_1(x, p) = f_2(y, q)$
 - 3.5 Charpit's method of solving nonlinear partial differential equations of first order
- 4 Modelling through Partial Differential Equation (10 hours)**
- 4.1 Second order partial differential equation and classification
 - 4.2 One-dimensional wave equation
 - 4.3 One-dimensional heat equation
 - 4.4 Two-dimensional heat equation, Laplace equation in Cartesian form
 - 4.5 Mass balance equation; equation of continuity in fluid dynamics, Navier-Stoke's equation
 - 4.6 Momentum balance equation; Euler's equation of motion for inviscid fluid flow
- 5 Z- transform and its Applications (6 hours)**
- 5.1 Representation of a sequence and basic operations
 - 5.2 Definition and existence of Z-transform, Z-transform of standard sequences
 - 5.3 Properties of Z-transform; linearity, change of scale, shifting properties, initial and final value theorems
 - 5.4 Differentiations of Z-transform
 - 5.5 Inverse Z-transform; partial fraction and residue methods
 - 5.6 Convolution of sequences, convolution of Z- transform
 - 5.7 Difference equations, application of Z-transform to solve difference equations and to find the sum of series
- Tutorial (30 hours)**
1. Problems related to find period and identify odd and even functions
 2. Exercises on Fourier series representation over intervals $2l$ and generalization into 2π
 3. Exercises related to Fourier series for discontinuous functions
 4. Exercises related to half range Fourier series
 5. Exercises related to complex form of Fourier series
 6. Exercises related to Fourier integral, Fourier sine and cosine integral
 7. Exercises related to Fourier transform, Fourier sine and cosine transform and inversion
 8. Exercises related to Fourier transform of derivatives and boundary value problems.

9. Exercises on application of C-R equations and construction of analytic functions
10. Exercises on application of analytic functions to flow problems
11. Exercises on mapping covering example of each mapping
12. Exercises on application of Cauchy integral theorem and formula
13. Exercises related to expansion of a function in Taylor and Laurent series
14. Exercises related to complex integration by using Cauchy's residue theorem
15. Exercises on solution of partial differential equation by direct integration
16. Exercises related to Lagrange's equation and PDE's as mentioned in 3.4
17. Exercises related to solution of one dimensional wave equation, one dimensional heat equation, two dimensional equation
18. Exercises related to Z-transform, application of properties
19. Exercises related to inverse Z-transform
20. Exercises related to solve difference equations by Z-transform
21. Exercises related to find sum of series by Z- transform

Final Exam

The questions will cover all the chapters in the syllabus. The evaluation scheme will be as indicated in the table below:

Chapter	Hours	Marks distribution*
1	12	18
2	12	18
3	5	6
4	10	10
5	6	8
Total	45	60

* There may be minor deviation in marks distribution.

References

1. Jeffery A. (2002). Advanced Engineering Mathematics (2nd edition). San Diego: Harcourt Academic Press.
2. O'Neill, P.V. (2011). Advanced Engineering Mathematics (7th edition). India: Thompsons, USA/Baba Baghanath Printers.
3. Kreyszig, A. (2020). Advanced engineering Mathematics (10th edition). USA: Wiley Publications.
4. Sastry S.S. (2014). Engineering Mathematics vol I and II (4th edition). India: PHI Learning Pvt. Ltd.
5. Wylie C., Barrett L. (1988). Advanced Engineering Mathematics (5th edition). McGraw Hill.
6. Dutta, D. (2006). A text book of Engineering Mathematics Vol I and II (2nd edition). India: New Age International Publishers.
7. Ogata, K. (2015). Discrete Time Control System (2nd edition). Pearson Publications.
8. Sharma, Sanjay. (2017). Signals and Systems (9th edition). India: S.K.Kataria and Sons.

COMMUNICATION ENGLISH

ENSH 204

Lecture : 3
Tutorial : 0
Practical : 1

Year : II
Part : I

Course Objectives:

The general objective of this course is to focus on English as a communication tool. Specifically, it emphasizes using English for professional communication for engineering works. It aims to increase English language ability to use appropriate research formats and methodology, develop concept papers, prepare research proposals and abstracts, set research questions, write a literature review, determine a research gap, link ideas, write technical proposals, prepare formal and informal reports and engage in project works, seminars/conferences.

- 1 Technical Communication (2 hours)**
 - 1.1 Definition, nature and scope of technical communication
 - 1.2 Professional ethics in communication (Ethical issues, plagiarism and copyright concerns, honesty, transparency and clarity)
- 2 Writing Skills (8 hours)**
 - 2.1 Principles of effective technical writing (Clarity, conciseness and coherence)
 - 2.2 Grammar (Pronoun and its antecedent, subject-verb agreement, non-finite verbs), sentence construction (Simple, compound, complex, and mixed sentences), error analysis and punctuation
 - 2.3 Bias-free language guideline, reducing bias
- 3 Technical Writing (15 hours)**
 - 3.1 Technical proposals (Purpose, types, structure, key considerations and examples)
 - 3.2 Research proposals and reports (Title page, table of contents, summary and abstract)
 - 3.3 Technical reports (Progress, feasibility and case study)
 - 3.4 Manuscript for journal (Structure, key considerations and examples)
 - 3.5 Citation and referencing (In-text citation, direct quote citations, indent citation, indirect citation, citing from books and journals, citing multiple authors in a single text, citing multiple texts from the same author, using numerical, pagination, preparing a reference page)

4 Business Correspondence (10 hours)

- 4.1 Writing formal letters (Applications, inquiries, complaints and orders)
- 4.2 E-mails (Structure, etiquette, and tone)
- 4.3 Notice, minutes and memos
- 4.4 Resume and cover letter
- 4.5 Press release/communique
- 4.6 Calling tender and responding to it

5 Listening and Oral Communication (4 hours)

- 5.1 Active listening (Barriers and strategies)
- 5.2 Effective speaking skills (Clarity, tone and pace)
- 5.3 Oral presentation skills (Structuring a presentation and handling questions)
- 5.4 Group discussions (Strategies and active participation)
- 5.5 Public speaking and speech delivery techniques

6 Use of Visual Aids in Communication (6 hours)

- 6.1 Tables
- 6.2 Graphs
- 6.3 Charts
- 6.4 Diagrams

Practical (15 hours)

- 1. Listening skill test
- 2. Visual skill test
- 3. Reading skill test
- 4. Oral communication test
- 5. Presentation skill test
- 6. Research proposals and project proposals
- 7. Team-based technical writing and presentations
- 8. Presentation on the prescribed texts
 - 8.1 "Which is More Important When Designing a Building: Beauty or Function?" from Unlock: Reading and Writing Skills by Chris Sowton
 - 8.2 On Being Modern-minded (Bertrand Russell)
 - 8.3 A Fable of Tomorrow" from The Silent Spring by Rachel Carson
 - 8.4 Religion and Science (From The World as I See It- Albert Einstein)
 - 8.5 "The Tamarisk Hunter" from Metatropolis by Paolo Bacigalupi
 - 8.6 Artificial Intelligence from The Art of Doing Science and Engineering by Richard W. Hamming
 - 8.7 Guglielmo Marconi and the History of Radio. Part II (Gerald A. Isted)
 - 8.8 Human-Centered Design (From The Design of Everyday Things- Don Norman)
 - 8.9 "The Paper Managerie" from The Paper Managerie and Other Stories by Ken Liu

8.10 "The Algorithm Will Save Us" from The New Voices of Fantasy by Sam J. Miller

8.11 "The Phantom Heart" by Laurence Yep

8.12 "Everyday Use" by Alice Walker

Final Exam

The questions will cover all the chapters in the syllabus. The evaluation scheme will be as indicated in the table below:

Chapter	Hours	Marks distribution*
1	2	5
2	8	10
3	15	20
4	10	10
5	4	10
6	6	5
Total	45	60

* There may be minor deviation in marks distribution.

References

1. Markel, M. and Selber, S. A. (2018). Technical communication (12th edition). Bedford/St. Martin's.
2. Ingre, D. (2017). Engineering communication: A practical guide to workplace communications for engineers (2nd edition). Cengage Learning.
3. Weisman, H. M. (2000). Technical communication for engineers: A handbook for engineers, scientists, and technicians. Prentice Hall.
4. Stevenson, S. and Whitmore, S. (2002). Strategies for engineering communication. John Wiley & Sons.
5. Rothwell, E. J., Cloud, M. J. (2017). Engineering writing by design: Creating formal documents of lasting value. CRC Press.
6. Blake, G., Bly, R. W. (1993). The elements of technical writing. Macmillan.
7. Beer, D., Mc Murrey, D. (2013). A guide to writing as an engineer (4th edition). John Wiley and Sons.
8. Farhathullah, T. M. (2002). Communication skills for technical students. Orient Longman.
9. Lebrun, J. L. (2007). Scientific writing: A reader and writer's guide. World Scientific Publishing.
10. Ligawa, H. (2021). Communication skills notes. Siaya Institute of Technology.
11. Katz, M. J. (2009). From research to manuscript: A guide to scientific writing (2nd edition). Springer.
12. Swales, J. M., Feak, C. B. (2012). Academic writing for graduate students: Essential tasks and skills (3rd edition). University of Michigan Press.
13. Hofmann, A. H. (2014). Scientific writing and communication: Papers, proposals, and presentations (2nd edition). Oxford University Press.

COMPUTER GRAPHICS AND VISUALIZATION

ENCT 201

Lecture : 3
Tutorial : 1
Practical : 3

Year : II
Part : I

Course Objectives:

The objective of this course is to provide basic principles and their applications to computer graphics and visualization. After completion of this course, students will be able to understand geometric transformations and their applications, two-dimensional and three-dimensional object modeling techniques, rendering techniques, animation and get an overview of the latest trends in computer graphics.

1 Introduction and Application (4 hours)

- 1.1 History of computer graphics
- 1.2 Overview of graphic systems
 - 1.2.1 Video display devices: Raster-scan displays, random-scan displays, flat panel displays, three-dimensional viewing devices
 - 1.2.2 Graphics software and tools: Coordinate representations, graphics functions, software standards, PHIGS workstations, DirectX, OpenGL, WebGL, Maya, Blender, Unity
- 1.3 Graphics pipeline
 - 1.3.1 Two-dimensional (2D) viewing pipeline
 - 1.3.2 Three-dimensional (3D) viewing pipeline
- 1.4 Applications in various fields like medicine, engineering, art, uses in augmented and virtual realism

2 Raster Graphics and Algorithms (9 hours)

- 2.1 Rasterizing a point
- 2.2 Rasterizing a straight line: DDA line algorithm, Bresenham's line algorithm
- 2.3 Rasterizing a circle and an ellipse: Mid-point circle and ellipse algorithm
- 2.4 Scan-line polygon fill algorithm
- 2.5 Scan-line fill of curved boundary areas
- 2.6 Boundary-fill algorithm
- 2.7 Flood-fill algorithm
- 2.8 Point clipping

- 2.9 Line clipping
 - 2.9.1 Cohen-Sutherland line clipping
 - 2.9.2 Liang-Barsky line clipping
- 2.10 Polygon clipping: Weiler-Atherton polygon clipping
- 2.11 Text clipping

3 2D and 3D Coordinate Systems and Viewing Transformations (9 hours)

- 3.1 2D transformation: Translation, rotation, scaling, reflection, shear
- 3.2 2D composite transformation
- 3.3 Window-to-viewport coordinate transformation
- 3.4 3D display methods
 - 3.4.1 Parallel projection
 - 3.4.2 Perspective projection
- 3.5 3D transformation: Translation, rotation, scaling, reflection, shear
- 3.6 3D composite transformation
- 3.7 Projection and viewing transformation

4 Curve Modeling and Surface Modelling (4 hours)

- 4.1 Introduction to parametric cubic curves, splines, Bezier curves
- 4.2 Surface modeling (Polygon surface, vertex table, edge table, polygon table, surface normal and spatial orientation of surfaces)

5 Visible Surface Determination (4 hours)

- 5.1 Image space and object space techniques
- 5.2 Back face detection, Z-Buffer, A-Buffer, Scan-Line method

6 Illumination and Surface Rendering Methods (4 hours)

- 6.1 Algorithms to simulate ambient, diffuse and specular reflections
- 6.2 Constant, Gouraud, Phong and Fast Phong shading models

7 Computer Animation and Visualization (5 hours)

- 7.1 Computer animation functions
- 7.2 Raster animations
- 7.3 Key-frame systems
- 7.4 Motion specifications
 - 7.4.1 Direct-motion specifications
 - 7.4.2 Goal-directed systems
 - 7.4.3 Kinematics and dynamics

8 Latest Trends in Computer Graphics

(6 hours)

- 8.1 Interactive visualization
- 8.2 Distributed scene rendering
- 8.3 Augmented reality (AR), virtual reality (VR) and mixed reality (MR)
- 8.4 Game development and real-time graphics
- 8.5 Applications of AR, VR and gaming

Tutorial

(15 hours)

1. Computations regarding raster graphics system- frame buffer size, color manipulation techniques, aspect ratio, refresh rate, resolution
2. Implementation of studied algorithms to determine points for digitizing lines, circles, and ellipses
3. Computational problems related to different clipping algorithms
4. Solution of problems related to 2D and 3D transformations and matrix compositions, including fixed-point scaling, pivot-point rotation, and reflection across an arbitrary line, among others
5. Transformation of object descriptions from the window coordinate system to the viewport coordinate system and solving problems related to parallel and perspective projection
6. Calculation of the points required to construct different curves studied using the specified set of control points and the desired number of line segments
7. Calculation of the surface normal of polygons and evaluating visibility using various visible surface determination techniques
8. Calculation of average intensity at a point on a polygon using Gouraud shading

Assignment

Appropriate assignment problems are given to students after the completion of each chapter.

Practical

(45 hours)

1. DDA line algorithm
2. Bresenham's line algorithm
3. Mid-point circle algorithm
4. Mid-point ellipse algorithm
5. Lab on 2-D transformations
6. Lab on 3-D transformations
7. Program for viewing and shading the 3D object
8. Clipping hidden surface removal
9. Implement the discrete techniques
10. Basic drawing techniques in OpenGL
11. A simple computer animation
12. A simple AR/VR scene or application using a framework (e.g., Unity, ARKit, Unreal Engine)

By the end of the practical, students required to develop a prototype project to demonstrate their understanding of computer graphics concepts. Students will work in teams and are encouraged to explore new programming languages or platforms (e.g., Unity, Unreal Engine, WebGL) to complete their project.

Final Exam

The questions will cover all the chapters in the syllabus. The evaluation scheme will be as indicated in the table below:

Chapter	Hours	Marks Distribution*
1	4	5
2	9	13
3	9	13
4	4	5
5	4	5
6	4	5
7	5	7
8	6	7
Total	45	60

* There may be minor deviation in marks distribution.

References

1. Hearn D., Baker, M. P. (1997). Computer Graphics C version (2nd edition), Prentice Hall.
2. Theoharis, T., Papaioannou, G., Platis, N., Patrikalakis, N. M. (2008). Graphics and Visualization: Principles & Algorithms. United States: CRC Press.
3. Foley, J. D. (1995). Computer Graphics: Principles and Practice in C (2nd edition). Germany: Addison-Wesley.

FOUNDATION OF DATA SCIENCE

ENCT 202

Lecture : 3
Tutorial : 1
Practical : 3

Year : II
Part : I

Course Objectives:

The objective of this course is to introduce the core concepts, tools, and methodologies of data science, focusing on the tools and techniques needed to analyze and interpret data effectively. Using data science tools, students will cover the entire data science process, from data acquisition, data manipulation, visualization, probability, statistics, and machine learning, with applications in business and engineering.

- 1 Introduction to Data Science (3 hours)**
 - 1.1 Overview of data science
 - 1.2 Jargons of data science
 - 1.3 Modern data ecosystem
 - 1.4 Data science lifecycle
 - 1.5 Trends, markets and applications of data science
 - 1.6 Tools and technologies in data science
 - 1.7 Data scientist and their roles
- 2 Mathematics for Data Science (10 hours)**
 - 2.1 Introduction to linear algebra for data science
 - 2.2 Vectors, matrices and matrix factorization
 - 2.3 Gradient descent for optimization
 - 2.4 Introduction to probability and random variable
 - 2.5 Probability distributions: Normal, Bernoulli, Binomial, Poisson
 - 2.6 Descriptive and inferential statistics
 - 2.7 Central limit theorem and sample distribution concepts
 - 2.8 Normal approximation; hypothesis testing procedures: Tests about the mean of a normal population
 - 2.9 The t-test, Z-tests for differences between two populations means, the two-sample t-test, confidence interval for mean of normal population
 - 2.10 ANOVA
- 3 Data Understanding and Preprocessing (10 hours)**
 - 3.1 Types of data: Structured, unstructured, semi-structured
 - 3.2 Data preprocessing requirements
 - 3.3 Data sources and collection methods

- 3.4 Data cleaning and preparation
- 3.5 Data wrangling and associated tools
- 3.6 Data enrichment, validation and publishing
- 3.7 Data transformation and normalization
- 3.8 Dimensionality reduction linear factor model, principal component analysis (PCA)

4 Data Analysis (8 hours)

- 4.1 Data analytics: Descriptive, diagnostic, predictive and prescriptive analytics
- 4.2 Exploratory data analysis using descriptive statistics
- 4.3 Data visualization
- 4.4 Data visualization techniques
- 4.5 Principles of effective data visualization
- 4.6 Feature engineering and other aspects of data manipulation

5 Regression and Predictive Modeling (5 hours)

- 5.1 Empirical models, simple linear regression, MLE and least square estimator
- 5.2 Multiple linear regression, matrix approach to multiple linear regression, polynomial regression models, categorical regressors, indicator variables, selection of variables and model building
- 5.3 Logistic regression

6 Modeling and Validation Processes (6 hours)

- 6.1 Introduction to machine learning
- 6.2 Introduction to supervised, unsupervised and reinforcement learning
- 6.3 Modeling process, training /validating model, cross validation methods, predicting new observations interpretation
- 6.4 Measures for model performance and evaluation: Classification accuracy, confusion matrix, sensitivity, specificity, precision, recall, F-score, ROC curve, clustering performance measures, other measures

7 Ethics and Recent Trends (3 hours)

- 7.1 Ethical considerations in data science
- 7.2 Data privacy regulations
- 7.3 Responsible data usage
- 7.4 The five Cs
- 7.5 Future trends

Tutorial (15 hours)

- 1. Solution of data problems using linear algebra, vectors and matrices
- 2. Solution of the problems related probability and statistics to understand application in data science

3. Identification of the data types and performing data cleaning, transformation, wrangling, and dimensionality reduction Including EDA and feature engineering
4. Solution of the problem related to linear and logistic regression
5. Understanding machine learning basics by model training, cross-validation, and performance evaluation

Practical

(45 hours)

1. Get acquainted with data science tools and perform statistical analysis
2. Hypothesis tests (e.g., t-tests, Z-tests) on sample datasets to compare population means
3. Simulate and apply the central limit theorem (CLT) to demonstrate how sample distributions converge to a normal distribution
4. Perform data wrangling and ETL processes on a dataset, followed by exploratory data analysis (EDA)
5. Utilize tools to create effective data visualizations (e.g., line charts, bar charts, heat maps, box plots) to derive key insights from the dataset
6. Implement feature extraction and selection techniques, including experimenting with encoding methods like one-hot encoding and creating new features based on domain expertise
7. Develop a simple linear regression model, extend it to multiple linear regression with several variables, and visualize both the regression line and residual plots
8. Apply logistic regression and evaluate the model using metrics such as accuracy, precision, recall, and the ROC curve
9. Apply K-means clustering and assess cluster quality using evaluation metrics like the silhouette score

By the end of the practical, students are required to submit a project where they develop a prototype to solve a real-world problem.

Final Exam

The questions will cover all the chapters in the syllabus. The evaluation scheme will be as indicated in the table below:

Chapter	Hours	Marks distribution*
1	3	6
2	10	12
3	10	12
4	8	9
5	5	6
6	6	9
7	3	6
Total	45	60

* There may be minor deviation in marks distribution.

References

1. Ozdemir, S. (2016). Principles of Data Science. Germany: Packt Publishing.
2. Maheshwari A. (2018). Data Science for Dummies, Wiley.
3. Grus, J. (2019). Data Science from Scratch: First Principles with Python. United States: O'Reilly Media.
4. Bruce, P., Bruce, A. (2017). Practical Statistics for Data Scientists: 50 Essential Concepts. United States: O'Reilly Media.
5. VanderPlas, J. (2016). Python Data Science Handbook: Essential Tools for Working with Data. United States: O'Reilly Media.
6. Provost, F., Fawcett, T. (2013). Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking. United States: O'Reilly Media.

THEORY OF COMPUTATION

ENCT 203

Lecture : 3
Tutorial : 1
Practical : 0

Year : II

Part : I

Course Objectives:

The objective of this course is to introduce students to the foundational concepts of theory of automata, formal languages, computational models and computational complexity.

- 1 Introduction to Formal Language, Logic and Proof (7 hours)**
 - 1.1 Brief review of set theory, function and relation
 - 1.2 Propositional logic, expressing statements in propositional logic, rules of inference and proofs in propositional logic, introduction to predicate logic
 - 1.3 Proofs, principle of mathematical induction, diagonalization principle, pigeonhole principle
 - 1.4 Alphabet and language
 - 1.5 Operations on languages: Union, concatenation, Kleene star
- 2 Finite Automata and Regular Language (10 hours)**
 - 2.1 Introduction to finite automata, finite state machine
 - 2.2 Deterministic finite automata (DFA), representation of DFA, language of DFA, design of DFA
 - 2.3 Non deterministic finite automata (NFA), equivalence of DFA and NFA
 - 2.4 Finite automata with epsilon transition (ϵ - NFA), equivalence of NFA and ϵ -NFA, equivalence of DFA and ϵ - NFA
 - 2.5 Regular expressions and regular languages
 - 2.6 Equivalence of regular expression and finite automata
 - 2.7 Closure properties of regular languages
 - 2.8 Pumping lemma for regular languages
 - 2.9 Decision algorithm for regular language
- 3 Context Free Grammar and Pushdown Automata (10 hours)**
 - 3.1 Introduction to context free grammar (CFG), component of CFG, context free language (CFL)
 - 3.2 Types of derivations, parse tree and its construction, ambiguity
 - 3.3 Simplification of CFG, normal forms, Chomsky normal form (CNF), Greibach normal form (GNF), Backus-Naur form (BNF)
 - 3.4 Closure properties of context free languages

- 3.5 Pumping Lemma for context free languages
- 3.6 Decision algorithm for context free language
- 3.7 Introduction to push down automata (PDA), representation of PDA, operations of PDA, move of a PDA, instantaneous description for PDA
- 3.8 Language of PDA, equivalence of CFL and PDA, conversion of CFG to PDA
- 3.9 Context sensitive grammar

4 Turing Machine (10 hours)

- 4.1 Introduction to turing machine (TM), representation of TM, move of a TM, instantaneous description for TM
- 4.2 Computing with turing machine
- 4.3 Variants of turing machine
- 4.4 Unrestricted grammar, Chomsky hierarchy of grammar
- 4.5 Recursive function theory

5 Decidability and Computational Complexity (5 hours)

- 5.1 Church turing thesis
- 5.2 Universal turing machine, encoding of turing machine
- 5.3 Undecidable problem about turing machines, halting problems and its implications
- 5.4 Computational complexity, time and space complexity of a turing machine
- 5.5 Complexity classes class P, class NP, NP-complete problems

6 Automata Theory and Compiler (3 hours)

- 6.1 Basic concept of compiler, role of lexical analyzer, lexical analysis with deterministic finite automata
- 6.2 Parser and context free grammar, top down parsing, bottom up parsing, LR parsing

Tutorial (15 hours)

- 1. Set operations and proof using mathematical induction
- 2. Proof using rules of inference in propositional logic
- 3. Design of DFA, conversion of NFA to DFA, proof using pumping lemma for regular language
- 4. Writing grammar for context free language, design of PDA, proof using pumping Lemma for context free language
- 5. Design of turing machine for a language
- 6. Problem related to compiler design

Final Exam

The questions will cover all the chapters in the syllabus. The evaluation scheme will be as indicated in the table below:

Chapter	Hours	Marks Distribution*
1	7	9
2	10	13
3	10	13
4	10	14
5	5	6
6	3	5
Total	45	60

* There may be minor deviation in marks distribution.

References

1. Lewis, H. R., Papadimitriou, C. H. (1981). Elements of the Theory of Computation. United Kingdom: Prentice-Hall.
2. Sipser, M. (2006). Introduction to the Theory of Computation. United Kingdom: Thomson Course Technology.
3. Rosen, K. (2006). Discrete Mathematics and Its Applications. United Kingdom: McGraw-Hill Education.
4. Aho, A. V. (2003). Compilers: Principles, Techniques and Tools (for VTU). India: Pearson.

MICROPROCESSORS

ENEX 201

Lecture : 3
Tutorial : 1
Practical : 3

Year : II
Part : I

Course Objectives:

The objective of this course is to familiarize students with assembly language programming, hardware and applications of microprocessors. The course provides students with a comprehensive understanding of microprocessor architecture, programming, and interfacing techniques, with a focus on the Intel 8085 and 8086 microprocessors. Students will gain theoretical and practical knowledge of microprocessor-based systems, covering essential topics such as computer organization, instruction sets, memory and I/O interfacing, interrupt mechanisms, and advanced architectural concepts.

1 Introduction (4 hours)

- 1.1 Introduction to microprocessors
- 1.2 History of microprocessors
- 1.3 Basic block diagram of a digital computer
- 1.4 Microcomputer and microcontroller
- 1.5 Bus organization of computer system
- 1.6 Stored program concept (Von Neumann's architecture)
- 1.7 Processing cycle of a stored program computer

2 Intel 8085 Microprocessor (12 hours)

- 2.1 Features of 8085
- 2.2 Internal architecture of 8085
- 2.3 Pin configurations of 8085
- 2.4 Instruction format and data format
- 2.5 Addressing modes of 8085
- 2.6 Instruction set of 8085
- 2.7 Various programs in 8085
 - 2.7.1 Simple programs with arithmetic and logical operations
 - 2.7.2 Conditions and loops
 - 2.7.3 Array and table processing
 - 2.7.4 Decimal - BCD conversion
 - 2.7.5 Multiplication and division
- 2.8 Instruction cycle, machine cycle and T-cycle
- 2.9 Timing diagrams of bus operations

3 Intel 8086 Microprocessor

(14 hours)

- 3.1 Features of 8086
- 3.2 Internal architecture of 8086
 - 3.2.1 BIU and components
 - 3.2.2 EU and components
 - 3.2.3 EU and BIU operations
- 3.3 Segment and offset address
- 3.4 Pin configurations of 8086
- 3.5 Addressing modes of 8086
- 3.6 Assembly language programming
- 3.7 High level versus low level programming
- 3.8 Assembly language syntax
 - 3.8.1 Comments
 - 3.8.2 Reserved words
 - 3.8.3 Identifiers
 - 3.8.4 Statements
 - 3.8.5 Directives
 - 3.8.6 Operators
 - 3.8.7 Instructions
- 3.9 EXE and COM programs
- 3.10 Assembling, linking and execution
- 3.11 One pass and two pass assemblers
- 3.12 Interrupt services: INT 21H and INT 10H
- 3.13 Various programs in 8086
 - 3.13.1 Simple programs for arithmetic, logical, string and input/output
 - 3.13.2 Conditions and loops
 - 3.13.3 Array and string processing
 - 3.13.4 ASCII and decimal numbers operation
 - 3.13.5 Displaying numbers in decimal, binary and hexadecimal formats

4 Microprocessor System

(7 hours)

- 4.1 Memory classifications and hierarchy
- 4.2 Interfacing I/O devices and memory with 8085
 - 4.2.1 Address decoding
 - 4.2.2 Unique and non-unique address decoding
 - 4.2.3 I/O mapped I/O and memory mapped I/O
 - 4.2.4 I/O address decoding with NAND and block decoders
 - 4.2.5 Memory address decoding
- 4.3 Parallel Interface
 - 4.3.1 Modes: Simple, wait, single handshaking and double handshaking
 - 4.3.2 Introduction to programmable peripheral interface (PPI)

- 4.4 Serial Interface
 - 4.4.1 Synchronous and asynchronous transmission
 - 4.4.2 Serial interface standards: RS232, USB
 - 4.4.3 Introduction to USART

5 Interrupt Operations (5 hours)

- 5.1 Interrupt and its importance
- 5.2 Polling vs interrupt
- 5.3 Types of interrupts
- 5.4 Interrupts in 8085 and interrupt instructions
- 5.5 Interrupt service routine
- 5.6 Interrupt processing in 8085
- 5.7 Using programmable interrupt controllers (PIC)
- 5.8 Interrupt in 8086 and interrupt vector table
- 5.9 Interrupt processing in 8086

6 Advanced Topics (3 hours)

- 6.1 Harvard architecture
- 6.2 Accumulator based and register based architecture
- 6.3 Hardwired and microprogrammed control unit
- 6.4 RISC and CISC
- 6.5 Introduction to multiprocessing/multitasking
- 6.6 Digital signal processor

Tutorial (15 hours)

- 1. Assembly language programming with 8085
- 2. Assembly language programming with 8086
- 3. Input/output and memory address decoding

Problems are solved, and students are assisted by the teacher to solve additional programming problems during tutorial classes.

Assignment

Appropriate assignment problems should be given to students after the completion of each chapter.

Practical (45 hours)

- 1. Familiarization to programming kit and program entry/execution in kit
- 2. 8085 assembly language programming with data transfer instructions
- 3. 8085 assembly language programming with arithmetic instructions
- 4. 8085 assembly language programming with logical instructions
- 5. 8085 assembly language programming with branching and stack instructions
- 6. Miscellaneous and practical programming with 8085 such as multiplication/division

7. Familiarization with DEBUG and entry/executing programs
8. 8086 assembly language programming with simple programs and assemble link and execute programs
9. 8086 assembly language programming for input and output using INT 21H service
10. 8086 assembly language programming for display using INT 10H service
11. 8086 assembly language programs for various conditions and I/O operations
12. Lab test

Final Exam

The questions will cover all the chapters in the syllabus. The evaluation scheme will be as indicated in the table below:

Chapter	Hours	Marks distribution*
1	4	6
2	12	15
3	14	18
4	7	12
5	5	6
6	3	3
Total	45	60

* There may be minor deviation in marks distribution.

References

1. Gaonkar, R. S. (2002). Microprocessor Architecture, Programming and Applications with the 8085. United Kingdom: Prentice Hall.
2. Abel, P. (2000). IBM PC Assembly Language and Programming (5th edition). United Kingdom: Prentice Hall.
3. Hall, D. V. (1999). Microprocessors and Interfacing: Programming and Hardware (2nd Edition). Tata McGraw Hill.
4. Stalling, W. (2009). Computer Organization and Architecture. Prentice Hall.



YEAR II, PART II

Semester 4

6 COURSES IN THIS SEMESTER

ENSH 252	Numerical Methods	76
ENEX 252	Instrumentation	80
ENEX 254	Electromagnetics	84
ENCT 252	Data Structure and Algorithm	87
ENCT 253	Data Communication	91
ENCT 254	Operating System	94



NUMERICAL METHODS

ENSH 252

Lecture : 3
Tutorial : 1
Practical : 3

Year : II

Part : II

Course Objectives:

The objective of this course is to equip students with a thorough understanding of numerical methods, focusing on their application in obtaining approximate solutions to complex mathematical problems commonly encountered in science and engineering. Emphasizing algorithm development, programming, and visualization techniques, the course enables students to apply computational approaches effectively, enhancing their problem-solving capabilities in real-world applications.

1 Solution of Non-Linear Equations (7 hours)

- 1.1 Errors and accuracy in numerical computations
- 1.2 Bisection method
- 1.3 Regula Falsi method and secant method
- 1.4 Newton Raphson method
- 1.5 Fixed point iteration method
- 1.6 Comparison of the methods (Bracketing vs open-ended methods and rates of convergence)
- 1.7 Solution of system of non-linear equations
 - 1.7.1 Direct approach
 - 1.7.2 Newton Raphson method

2 Solution of System of Linear Algebraic Equations (8 hours)

- 2.1 Direct methods
 - 2.1.1 Gauss Jordan method
 - 2.1.2 Gauss elimination method, pivoting strategies (Partial and complete)
 - 2.1.3 Matrix inverse using Gauss Jordan and Gauss elimination methods
 - 2.1.4 Factorization methods (Do-Little's method and Crout's method)
- 2.2 Iterative methods
 - 2.2.1 Jacobi's method
 - 2.2.2 Gauss-Seidal method
- 2.3 Determination of largest and smallest Eigen values and corresponding vectors using the power method

3 Interpolation (9 hours)

- 3.1 Polynomial Interpolation
 - 3.1.1 Finite differences (Forward, backward, central and divided differences)
 - 3.1.2 Interpolation with equally spaced intervals: Newton's forward and backward difference interpolation, Stirling's and Bessel's central difference interpolation
 - 3.1.3 Interpolation with unequally spaced intervals: Newton's divided difference interpolation, Lagrange interpolation
- 3.2 Least square method of curve fitting
 - 3.2.1 Linear form and forms reducible to linear form
 - 3.2.2 Quadratic form and forms reducible to quadratic form
 - 3.2.3 Higher degree polynomials
- 3.3 Cubic spline interpolation
 - 3.3.1 Equally spaced interval
 - 3.3.2 Unequally spaced interval

4 Numerical Differentiation and Integration (6 hours)

- 4.1 Numerical differentiation
 - 4.1.1 Differentiation using polynomial interpolation formulae for equally spaced intervals
 - 4.1.2 Local maxima and minima from equally spaced data
- 4.2 Numerical integration
 - 4.2.1 Newton Cote's general quadrature formula
 - 4.2.2 Trapezoidal rule, Simpson's 1/3 and 3/8 rules, Boole's rule, Weddle's rule
 - 4.2.3 Romberg integration
 - 4.2.4 Gauss-Legendre integration (up to 3-point formula)

5 Solution of Ordinary Differential Equations (ODE) (8 hours)

- 5.1 Initial value problems
 - 5.1.1 Solution of first order equations: Taylor's series method, Euler's method, Runge-Kutta methods (Second and fourth order)
 - 5.1.2 Solution of system of first order ODEs via Runge-Kutta methods
 - 5.1.3 Solution of second order ODEs via Runge-Kutta methods
- 5.2 Two-point boundary value problems
 - 5.2.1 Shooting method
 - 5.2.2 Finite difference method

6 Solution of Partial Differential Equations

(7 hours)

- 6.1 Introduction and classification
- 6.2 Finite difference approximations of partial derivatives
- 6.3 Solution of elliptic equations
 - 6.3.1 Laplace equation
 - 6.3.2 Poisson's equation
- 6.4 Solution of parabolic and hyperbolic equations
 - 6.4.1 One-dimensional heat equation: Bendre-Schmidt method, Crank-Nicolson method
 - 6.4.2 Solution of wave equation

Tutorial

(15 hours)

- 1. Solution of non-linear equations
- 2. Solution of system of linear algebraic equations
- 3. Polynomial interpolation
- 4. Least square method of curve fitting
- 5. Cubic spline interpolation
- 6. Numerical differentiation
- 7. Numerical Integration
- 8. Solution of ordinary differential equations (Initial value problems)
- 9. Solution of ordinary differential equations (Boundary value problems)
- 10. Solution of partial differential equations

Practical

(45 hours)

Programming language to be used: Python

Results to be visualized graphically wherever possible

Practical report contents: Working principle, Pseudocode, Source code, Test Cases

- 1. Basics of programming in Python:
 - Basic input/output
 - Basic data types and data structures
 - Control flow
 - Functions and modules
 - Basic numerical and scientific computation
 - Graphical visualization
- 2. Solution of Non-linear equations:
 - Bisection method
 - Secant method
 - Newton-Raphson
 - System of non-linear equations using Newton-Raphson method
- 3. System of linear algebraic equations:
 - Gauss Jordan Method
 - Gauss elimination method with partial pivoting
 - Gauss-Seidal method
 - Power method

4. Interpolation
 - Newton's forward difference interpolation
 - Lagrange interpolation
 - Least square method for linear, exponential and polynomial curve fitting
 - Cubic spline interpolation
5. Numerical Integration
 - Trapezoidal rule
 - Simpson's 1/3 rule or Simpson's 3/8 rule
 - Boole's Rule or Weddle's Rule
 - Gauss-Legendre integration
6. Solution of Ordinary Differential Equations:
 - Runge-Kutta fourth order method for first order ODE
 - Runge-Kutta fourth order method for system of ODEs / 2nd order ODE
 - Solution of two-point boundary value problem using Shooting method
 - Solution of two-point boundary value problem using finite difference method
7. Solution of partial differential equations using finite difference approach:
 - Laplace equation using Gauss-Seidal iteration
 - Poisson's equation using Gauss-Seidal iteration
 - One-dimensional heat equation using Bendre-Schmidt method
 - One-dimensional heat equation using Crank-Nicholson method

Final Exam

The questions will cover all the chapters in the syllabus. The evaluation scheme will be as indicated in the table below:

Chapter	Hours	Marks distribution*
1	7	10
2	8	10
3	9	10
4	6	10
5	8	10
6	7	10
Total	45	60

* There may be minor deviation in marks distribution.

References

1. Chapra, S. C., Canale, R. P. (2010). Numerical Methods for Engineers (6th edition). McGraw-Hill.
2. Kiusalaas, J. (2013). Numerical Methods in Engineering with Python 3 (3rd edition). Cambridge University Press.
3. Grewal, B. S. (2017). Numerical Methods in Engineering & Science (11th edition). India: Khanna Publishers.
4. Yakowitz, S., Szidarovszky, F. (1986). An Introduction to Numerical Computations (2nd edition). Macmillan Publishing.
5. Kong, Q., Siau T., Bayen A. (2020). Python Programming and Numerical Methods. Academic Press.

INSTRUMENTATION

ENEX 252

Lecture : 4
Tutorial : 1
Practical : 3/2

Year : II

Part : II

Course Objectives:

The objective of this course is to provide comprehensive understanding on methods and instrument for a wide range of measurement problems used in instrumentation system. It also covers application of transducers in the microprocessor, microcontroller and their interfacing to design instrumentation system.

1 Introduction (2 hours)

- 1.1 Analog and digital instrument: Definition, block diagram, characteristics
- 1.2 Microprocessor-based systems: Open vs closed loop, benefits, features and applications in instrumentation design
- 1.3 Microcomputer on instrumentation design

2 Theory of Measurement (6 hours)

- 2.1 Static performance parameters: Accuracy, precision, sensitivity, resolution and linearity
- 2.2 Dynamic performance parameters: Response time, frequency response and bandwidth
- 2.3 Error in measurement
- 2.4 Statistical analysis of error in measurement
- 2.5 Measurement of resistance (Low, medium and high)
- 2.6 DC / AC bridge (Wheatstone bridge, Maxwell's bridge, Schering bridge)

3 Transducer (8 hours)

- 3.1 Transducer, workflow of a transducer in typical system, transducer classification
- 3.2 Sensor and its working principle (Resistive, capacitive and piezoelectric), generation of sensor, classification of sensor (Analog sensor, digital sensor)
- 3.3 Types of sensors (Electrical sensor, chemical sensor, biological sensor, acoustic sensor, optical sensor and other motion sensor), characteristic of sensors
- 3.4 Actuator, classification of actuators (Hydraulic, pneumatic, electric and mechanical), characteristic of actuator

- 4 Interfacing of Instrumentation System (14 hours)**
- 4.1 Microprocessor and microcontroller and their selection criteria, and applications
 - 4.2 The PPI 8255 and interfacing of peripherals (LED, 7 segment, dip switch, 8-bit ADC, 8/10-bit DAC using mode 0 and mode1) with 8085 microprocessor
 - 4.3 Microcontrollers (Atmega328, STM32): Architecture, pin configuration, and their application
 - 4.4 Sensor/Actuator interfacing with Atmega328P (Arduino): Analog and digital sensors, implementation of communication protocols, interrupt based interfacing
- 5 Connectivity Technology in Instrumentation System (6 hours)**
- 5.1 Wired and wireless communication system
 - 5.2 Wired connectivity: UART, I2C, SPI, CAN
 - 5.3 Wireless sensor network and its technology
 - 5.4 RF modem, Bluetooth, WI-FI, NFC, ZIGBEE and LORA
 - 5.5 Thermal management: Heat dissipation technique, heat sink
 - 5.6 Data acquisition system (Data loggers, data archiving and storage), cloud based data acquisition system
- 6 Circuit Design (4 hours)**
- 6.1 Converting requirement into design, reliability and fault tolerance
 - 6.2 High-speed design: Bandwidth, decoupling, crosstalk, impedance matching
 - 6.3 PCB design: Component placement, trace routing, signal integrity, and ground loops
 - 6.4 Noise and noise coupling mechanism, noise prevention, filtering, ferrite beads, decoupling capacitors, and ESD & its prevention
- 7 Software for Instrumentation Application (6 hours)**
- 7.1 Overview of software engineering
 - 7.2 Types of software
 - 7.3 Software development life cycle (SDLC), software process models (Waterfall model, prototype model, incremental model, agile model)
 - 7.4 Software reliability vs hardware reliability
 - 7.5 Software bugs, software testing, different levels of testing
- 8 Electrical Equipment (6 hours)**
- 8.1 Voltmeter and ammeter: Types and working principle
 - 8.2 Energy meter: Types and working principle
 - 8.3 Frequency meter: Types and working principle
 - 8.4 Wattmeter: Types and working principle

9 Latest Trends

(3 hours)

- 9.1 Internet of things (IoT): Simple architecture, characteristics, advantages
- 9.2 Smart sensors
- 9.3 Important of cloud computing in instrumentation system
- 9.4 Instrumentation in industry 4.0/5.0

10 Application of Modern Instrumentation System

(5 hours)

- 10.1 Instrumentation for power station including all electrical and non-electrical parameters
- 10.2 Instrumentation for wire and cable manufacturing and bottling plant
- 10.3 Instrumentations for a beverage manufacturing and bottling plant
- 10.4 Instrumentations required for a biomedical application such as a medical clinic or hospital
- 10.5 Instrumentation system design using a processor (Microprocessor, microcontroller or others)

Tutorial

(15 hours)

1. Understanding the fundamentals of Op-amps is essential since they are central to analog instrumentation.
2. Learn how to filter, amplify, and modify analog signals for signal conditioning
3. How ADCs and DACs work and how to select the right one for your application
4. Interfacing of ADC on any application of your interest
5. Application for the protocol UART, I2C, SPI in Adriano
6. Design a simple temperature sensing system using a thermistor or thermocouple, op-amp, and analog display.
7. Explain the generation of PWM signals in ATmega328P for controlling things like motor speed or LED brightness.

Practical

(22.5 hours)

1. Measurement and accuracy testing: Analog and digital meters
2. Use of LabVIEW, Proteus, MATLAB or others for modeling instrumentation systems
3. Use of resistive, capacitive & inductive transducers / sensors / actuators
4. Review of assembly programming and simple I/O interfacing with 8085 and 8255
5. Interfacing of LEDs, seven segment display and motors
6. Interfacing of ADC and DAC

Final Exam

The questions will cover all the chapters in the syllabus. The evaluation scheme will be as indicated in the table below:

Chapter	Hours	Marks distribution*
1	2	4
2	6	5
3	8	6
4	14	12
5	6	6
6	4	4
7	6	6
8	6	6
9	3	5
10	5	6
Total	60	60

* There may be minor deviation in marks distribution.

References

1. Hall, D. V., (1999). Microprocessor and Interfacing, Programming and Hardware. Tata McGraw Hill
2. Goankar, R. S., (2000). Microprocessor Architecture, Programming and Application with 8085. Prentice Hall
3. Fowler, K. R., (1996). Electronic Instrument Design: Architecting for the Life Cycle. Oxford University Press, Inc.
4. Sawhney, A. K., (1998). A Course in Electronic Measurement and Instrumentation. Dhanpat Rai and Sons.
5. Gupta, J. B., (2008). A Course in Electrical and Electronics Measurement and Instrumentation, Kataria and Sons.
6. DE Silva C. W., Sensors and Actuators: Control System Instrumentation. CRC Press Taylor and French Group Boca Raton London New York.
7. Misra, S., Roy, C. and Mukherjee, A., (2020). Introduction to Industrial Internet of Things and Industry 4.0. CRC Press.

ELECTROMAGNETICS

ENEX 254

Lecture : 3
Tutorial : 1
Practical : 3/2

Year :II
Part : II

Course Objectives:

The objective of this course is to provide students with a basic mathematical concepts related to electromagnetic time invariant and time variant fields including electromagnetic wave and their transmission on different media

1 Introduction (4 hours)

- 1.1 Scalar and vector fields
- 1.2 Operations on scalar and vector fields
- 1.3 Co-ordinate systems (Cartesian, cylindrical and spherical) and conversions

2 Electric Field (15 hours)

- 2.1 Coulomb's law
- 2.2 Electric field intensity
- 2.3 Electric flux density
- 2.4 Gauss's law and applications
- 2.5 Physical significance of divergence, divergence theorem
- 2.6 Electric potential, potential gradient
- 2.7 Energy density in electrostatic field
- 2.8 Electric properties of material medium
- 2.9 Free and bound charges, polarization, relative permittivity, electric dipole electric boundary conditions
- 2.10 Current, current density, conservation of charge, continuity equation, relaxation time
- 2.11 Boundary value problems, Laplace and Poisson equations and their solutions, uniqueness theorem

3 Magnetic Field (9 hours)

- 3.1 Biot-Savart's law
- 3.2 Magnetic field intensity
- 3.3 Ampere's circuital law and its application
- 3.4 Magnetic flux density
- 3.5 Physical significance of curl, Stoke's theorem
- 3.6 Scalar and magnetic vector potential
- 3.7 Magnetic properties of material medium

- 3.8 Magnetic force, magnetic torque, magnetic moment, magnetic dipole, magnetization
- 3.9 Magnetic boundary condition

4 Time Varying Fields (4 hours)

- 4.1 Faraday's law, transformer EMF, motional EMF
- 4.2 Displacement current
- 4.3 Maxwell's equations in integral and point forms

5 Plane Waves (9 hours)

- 5.1 Wave propagation in lossless and lossy dielectric
- 5.2 Plane waves in free space, lossless dielectric, good conductor
- 5.3 Power and poynting theorem average power density
- 5.4 Reflection of plane wave at normal incidence
- 5.5 Standing wave and SWR
- 5.6 Input intrinsic impedance

6 Transmission Lines (4 hours)

- 6.1 Transmission line equations (Taking analogy from wave equations)
- 6.2 Lossless, lossy and distortionless transmission lines
- 6.3 Input impedance, reflection coefficient, standing wave ratio

Tutorial (15 hours)

1. Conversion of coordinate systems (Cartesian to cylindrical /spherical and vice versa, cylindrical to spherical and vice versa)
2. Electric field intensity and flux density (Coulomb's law, Gauss law, divergence, electric potential and energy density)
3. Boundary condition, electric dipole, and boundary value problems
4. Magnetic fields (Biot-Savart law, Ampere circuit law, Stoke's theorem, magnetic force and torque)
5. Time varying fields (Transformer/motional EMF, displacement current)
6. Wave propagation equations in lossy and lossless medium (Poynting theorem, standing wave ratio and intrinsic impedance)
7. Transmission line (Lossless, lossy and distortionless)

Practical (22.5 hours)

1. Teledeltos (Electro-conductive) paper mapping of electrostatic fields
2. Determination of dielectric constant, display of a magnetic hysteresis loop
3. Studies of wave propagation on a lumped parameter transmission line
4. Microwave sources, detectors, transmission lines
5. Standing wave patterns on transmission lines, reflections, power patterns on transmission lines, reflections, power measurement
6. Familiarizations of electric and magnetic field measurements using simulation tool

Final Exam

The questions will cover all the chapters in the syllabus. The evaluation scheme will be as indicated in the table below:

Chapter	Hours	Marks distribution*
1	4	5
2	15	20
3	9	12
4	4	6
5	9	12
6	4	5
Total	45	60

* There may be minor deviation in marks distribution.

References

1. Hayt, W. H. (2001). Engineering Electromagnetics. McGraw-Hill Book Company.
2. Kraus, J. D. (1973). Electromagnetics. McGraw-Hill Book Company.
3. Rao, N. N. (1990). Elements of Engineering Electromagnetics. Prentice Hall.
4. Devid K. Cheng, (1989). Field and Wave Electromagnetics. Addison-Wesley.
5. Sadiku, M. N. O. (2010). Elements of Electromagnetics. Oxford University Press.

DATA STRUCTURE AND ALGORITHMS

ENCT 252

Lecture : 3
Tutorial : 1
Practical : 3

Year : II
Part : II

Course Objectives:

The objective of this course is to impart fundamental knowledge on the design and implementation of data structures for storing information. It also covers various algorithms used in computer science. Upon completion of this course, students will be able to design and choose the appropriate data structure and efficient algorithm to achieve optimal performance

1 Introduction (4 hours)

- 1.1 Introduction to data structures
 - 1.1.1 Need of data structures
 - 1.1.2 Types of data structures and its characteristics
- 1.2 Abstract data type (ADT)
- 1.3 Basics of algorithm design techniques (Brute Force, divide and conquer, Greedy algorithms, branch and bound, backtracking, randomized, recursive, dynamic programming)
- 1.4 Algorithm analysis
 - 1.4.1 Time and space complexity
 - 1.4.2 Best, worst and average case analysis
 - 1.4.3 Rate of growth
 - 1.4.4 Asymptotic notations: Big Oh, Big Omega and Big Theta

2 Stack and Recursion (7 hours)

- 2.1 Definition of stack and its operations
- 2.2 Array implementation of stack ADT
- 2.3 Stack applications
 - 2.3.1 Expression conversion: Infix to postfix and prefix expression
 - 2.3.2 Expression evaluation: Infix and postfix expression evaluation
- 2.4 Recursion
 - 2.4.1 Concept of recursion
 - 2.4.2 Recursion and stack
 - 2.4.3 Recursion vs iteration
 - 2.4.4 Execution of recursive calls
 - 2.4.5 Types of recursions
 - 2.4.6 Applications of recursion: Tower of Hanoi

- 3 Queues (5 hours)**
- 3.1 Definition of queue and its operations
 - 3.2 Array implementation of queue ADT
 - 3.3 Types of queue ADT: Linear, circular, double ended and priority queues
- 4 Linked List (6 hours)**
- 4.1 Definition of list and its operations
 - 4.2 Array implementation of list ADT
 - 4.3 Static list and its limitations
 - 4.4 Linked list: Definition and its operations
 - 4.5 Types of linked list: Singly, doubly, circular
 - 4.6 Application of linked list
 - 4.6.1 Linked list implementation of stack and queue ADT
 - 4.6.2 Solving polynomial equations using linked list
- 5 Tree (7 hours)**
- 5.1 Definition and tree terminologies
 - 5.2 Binary trees
 - 5.2.1 Definition and types
 - 5.2.2 Array and linked list representation
 - 5.2.3 Traversal algorithms: Pre-order, in-order and post-order traversal
 - 5.2.4 Application of full binary tree: Huffman algorithm
 - 5.3 Binary search tree
 - 5.3.1 Definition and operations on binary search tree: Insertion, deletion, searching and traversing
 - 5.3.2 Construction of binary search tree
 - 5.4 Balanced binary tree
 - 5.4.1 Problem with unbalanced binary trees
 - 5.4.2 Balanced binary search tree
 - 5.4.3 AVL tree, definition and need of AVL tree, construction of AVL tree: Insertion, deletion on AVL tree and rotation operations
 - 5.5 Introduction to red-black tree
 - 5.6 B-Tree: Need, definition and construction of B-tree
- 6 Graphs (6 hours)**
- 6.1 Definition, terminologies and types of graphs
 - 6.2 Representation of graphs: Adjacency matrix, incidence matrix and adjacency list
 - 6.3 Transitive closure and Warshall's algorithm
 - 6.4 Graph traversals: Breadth-first search, depth-first search and topological sort

- 6.5 Minimum spanning tree: Kruskal's algorithm and prim's algorithm
- 6.6 Shortest-paths problems: Dijkstra's algorithm, Floyd- Warshall algorithm

7 Sorting Algorithms (5 hours)

- 7.1 Definition of sorting and its applications
- 7.2 Types of sorting: Internal/external sort, stable/unstable sort, in-place/ not in-place sort, adaptive/ non-adaptive sort
- 7.3 Sorting algorithms and its efficiency: Bubble, insertion, selection, shell, quick, merge, radix and heap sorting

8 Searching Algorithms (5 hours)

- 8.1 Definition of searching techniques and its applications
- 8.2 Different searching algorithms and its efficiency
 - 8.2.1 Sequential search
 - 8.2.2 Binary search
- 8.3 Hashing
 - 8.3.1 Definition and its applications
 - 8.3.2 Hash function
 - 8.3.3 Hash table
 - 8.3.4 Collision in hash table
 - 8.3.5 Collision resolution techniques: Chaining method and open addressing method (Linear probing, quadratic probing and double hashing)

Tutorial (15 hours)

- 1. Analyzing time and space complexity of basic algorithms and comparing best, worst, and average cases with examples
- 2. Solving problems using stacks: Converting infix expressions to postfix/prefix, evaluating postfix expressions, balancing parentheses, reversing a string
- 3. Solving Tower of Hanoi recursively and analyzing tail vs non-tail recursion
- 4. Solving polynomial addition using linked lists
- 5. Implementing tree traversals: Preorder, inorder, postorder
- 6. Constructing a binary search tree (BST) and performing insertion, deletion, and searching, balancing BSTs with AVL tree operations (Rotations, insertion, deletion), constructing B-Trees (Insertion, deletion)
- 7. Implementing Huffman coding for text compression
- 8. Solving shortest path problems using Dijkstra's and Floyd-Warshall algorithms
- 9. Implementing Kruskal's and Prim's algorithms for minimum spanning tree
- 10. Analyzing and comparing sorting algorithm efficiencies
- 11. Implementing linear search and binary search
- 12. Designing a hash table with different collision resolution techniques (Chaining, linear probing, quadratic probing, double hashing)

Practical**(45 hours)**

1. Implementation of stack using array, applications of stack- conversion of infix to prefix and postfix expression, evaluation of prefix and postfix expression, matching parenthesis, reversal of string
2. Implementation of recursive algorithms (Tail and non-tail method)- Factorial, sum of natural numbers, Fibonacci series, implementation of Tower of Hanoi
3. Implementations of linear queue and circular queue using arrays
4. Implementation of static list, implementation of linked list: Singly and doubly linked lists
5. Implementation of stack and queue using linked list, application of linked list- polynomial addition
6. Implementation of in-order, pre-order and post-order tree traversals
7. Implementation of breadth-first and depth-first search to traverse a graph
8. Implementation of different sorting algorithms
9. Implementation of different searching algorithms

Final Exam

The questions will cover all the chapters in the syllabus. The evaluation scheme will be as indicated in the table below:

Chapter	Hours	Marks distribution*
1	4	5
2	7	9
3	5	7
4	6	8
5	7	9
6	6	8
7	5	7
8	5	7
Total	45	60

* There may be minor deviation in marks distribution.

Reference

1. Langsam, Y. Augenstein .M. J. and Tenenbaum A. M. (1996). Data Structures using C and C++. Prentice Hall Press.
2. Rowe, G. W. (1997). Introduction to data structures and algorithms with C++. Prentice-Hall, India.
3. Cormen, T. H., Leiserson, C. E., Rivest, R. L., and Stein, C, (2022). Introduction to algorithms. MIT press.
4. Kruse, R. L., and Ryba, A. J., (1998). Data structures and program design in C++. Prentice Hall, India.
5. Thareja, R. (2014). Data Structures Using C. Oxford University Press.

DATA COMMUNICATION

ENCT 253

Lecture : 3
Tutorial : 1
Practical : 3/2

Year : II
Part : II

Course Objectives:

The objective of this course is to provide students with a solid foundation in the principles and theories of data communication, including key terminology, protocols, and standards. Also support to explore the various types of transmission media, including guided and unguided media, and their characteristics, advantages, and disadvantages. Furthermore, it introduces the methods of data encoding, modulation techniques, and their implications for effective data transmission.

1 Introduction (4 hours)

- 1.1 Analog data communication, data representation, data flows
- 1.2 Evolution of data communication
- 1.3 A communication model, data communication model
- 1.4 Networks (LAN, WAN), simplified network architecture, the OSI model
- 1.5 Data communication and networking for today enterprise

2 Data Communication Fundamentals (6 hours)

- 2.1 Analog and digital data
- 2.2 Analog signals, periodic and aperiodic signals, periodic signals characteristics (Time, frequency domain)
- 2.3 Introduction to Fourier series representation of periodic signal, Fourier transform representation of aperiodic signals, digital signals and its characteristics
- 2.4 Analog and digital transmission, transmission mode, transmission impairments (Attenuation, distortion, noise)
- 2.5 Data rate limits channel capacity, Nyquist bandwidth, Shannon capacity formula
- 2.6 Performance of network (Bandwidth, throughput, latency, jitter)

3 Transmission Media and Data Compression (8 hours)

- 3.1 Guided transmission media: Co-axial cable, twisted pair, optical fiber
- 3.2 Unguided transmission media: Radio waves, microwaves, infrared
- 3.3 Antenna basics, satellite communication, Bluetooth, Wi-Fi

- 3.4 Wireless propagation (Introduction to groundwave propagation, sky wave propagation and line of sight propagation), frequency bands
- 3.5 Error detection and correction: Parity, check sum, cyclic redundancy check, hamming code
- 3.6 Data compression: Lossy and lossless

4 Signal Encoding Technique (15 hours)

- 4.1 Analog data, analog signals: Modulation and its need, AM, FM, PM
- 4.2 Analog data, digital signals: PAM, PWM, PPM, PCM, DPCM, DM
- 4.3 Digital data, analog signal: ASK, FSK, PSK, QPSK, QAM
- 4.4 Digital data, digital signal: RZ, NRZ, AMI, Manchester, differential Manchester, B8ZS, HDB3 for data transmission

5 Multiplexing and Switching (8 hours)

- 5.1 Access introduction to multiplexing, application of multiplexing
- 5.2 Frequency division multiple
- 5.3 Time division multiple access
- 5.4 Asymmetric digital subscriber line, XDSL
- 5.5 Spread spectrum: DHSS, FHSS, CDMA
- 5.6 Intro switched communication network, connection oriented and connectionless
- 5.7 Switching devices: Types, importance and application
- 5.8 Circuit switching network: Circuit switching concepts, message switching
- 5.9 Packet switching: Virtual switching, datagram switching

6 Cellular Wireless Communications and Latest Trends (4 hours)

- 6.1 Overview of 1G, 2G, 3G and 4G
- 6.2 Cellular technology fundamental terminology: Cell, frequency-reuse, cluster, adjacent cell interference, co-channel interference, handoff strategies, architecture of GSM basics
- 6.3 Introduction to 5G networks, software defined networking, IOT communication, cloud computing and virtualization in data communication

Tutorial (15 hours)

- 1. Tutorials on different protocols in data communication TCP/IP, HTTP/HTTPS, FTP
- 2. Explore the function of open systems interconnection (OSI) model, which defines seven layers of data communication
- 3. Discover data communication devices and its application
- 4. Identify the application of used network topologies in present scenario
- 5. Collecting ideas on some security aspects of the security on data communication on present enterprises system

Practical**(22.5 hours)**

1. Signal analysis using MATLAB
2. Implementation of small network using hub and switch in physical or simulation environment
3. Analog modulation generation and reconstruction
4. Pulse modulation generation and reconstruction
5. Conversion of given binary sequence into different line coding
6. Digital modulation (ASK, FSK, PSK) generation and reconstruction

Final Exam

The questions will cover all the chapters in the syllabus. The evaluation scheme will be as indicated in the table below:

Chapter	Hours	Mark distribution*
1	4	5
2	6	9
3	8	11
4	15	20
5	8	10
6	4	5
Total	45	60

* There may be minor deviation in marks distribution.

Reference

1. Stallings, W. (2007). Data and Computer Communications. India: Pearson Education.
2. Forouzan, B. A., Fegan, S. C. (2007). Data Communications and Networking (McGraw-Hill Forouzan Networking). United Kingdom: McGraw-Hill Higher Education.
3. Tanenbaum, A. S., Wetherall, D. (2011). Computer Networks. India: Pearson Prentice Hall.
4. Rappaport, T. S. (2024). Wireless Communications: Principles and Practice. United Kingdom: Cambridge University Press.

OPERATING SYSTEM

ENCT 254

Lecture : 3
Tutorial : 1
Practical : 3/2

Year : II
Part : II

Course Objectives:

The objective of this course is to familiarize students with the different aspects of operating systems and encourage them to use these ideas in designing operating systems.

1 Introduction (6 hours)

- 1.1 Introduction to operating systems
- 1.2 OS as an extended machine and resource manager
- 1.3 History of operating system
- 1.4 Type of operating system: Mainframe, server, personal, smartphone and handheld, IOT and embedded, real-time, smart-card
- 1.5 Operating system components: Kernel, shell, utilities, applications
- 1.6 Types of OS kernel: Monolithic, micro, nano, layered, hybrid, exo-kernel
- 1.7 System calls, shell commands, shell programming
- 1.8 POSIX standard
- 1.9 Bootloader, MBR/GPT, UEFI and legacy boot

2 Process Management (7 hours)

- 2.1 Process description, states and control
- 2.2 Scheduling algorithms
 - 2.2.1 First Come First Serve (FCFS)
 - 2.2.2 Shortest Job First (SJF)
 - 2.2.3 Shortest Remaining Time(SRT)
 - 2.2.4 Round Robin (RR)
 - 2.2.5 Highest Response Ratio Next (HRNN)
 - 2.2.6 Completely Fair Scheduler (CFS) used in Linux
- 2.3 Threads and thread scheduling

3 Process Communication and Synchronization (10 hours)

- 3.1 Principles of concurrency, race condition, critical region
- 3.2 Mutual exclusion, semaphores, and mutex
- 3.3 Message passing and monitors
- 3.4 Classical problems of synchronization: Readers-writers problem, producer-consumer problem, dining philosopher problem
- 3.5 Deadlock: Prevention, ignorance, avoidance, detection and recovery

4 I/O and Memory Management (9 hours)

- 4.1 I/O management
 - 4.1.1 Principles of I/O hardware and software
 - 4.1.2 I/O software layer
 - 4.1.3 Disk technologies: Magnetic disk, SSD, NVMe storage
 - 4.1.4 RAID
 - 4.1.5 Concept of stable storage, cost per bit comparison
- 4.2 Memory Management
 - 4.2.1 Memory address, swapping and managing free memory space
 - 4.2.2 Virtual memory management, paging, segmentation
 - 4.2.3 Page replacement algorithms (FIFO, LRU, LFU), page fault and hit ratio
 - 4.2.4 Allocation of frames
 - 4.2.5 Thrashing

5 File Systems (3 hours)

- 5.1 File concepts: Name, structure, types, access, attributes, operations
- 5.2 Directory structures: Paths and hierarchies (Linux/Windows)
- 5.3 File system implementation: Inodes, allocation methods (Contiguous, linked, indexed)
- 5.4 File system performance: Factors affecting efficiency
- 5.5 Example file systems: NTFS, EXT4, FAT32, NFS

6 Security and System Administration (3 hours)

- 6.1 OS security: Cryptography, multi-factor authentication (MFA), secure boot and sandboxing
- 6.2 Access control: Policies, lists, and OS support
- 6.3 System administration: User management, environment setup and tools (AWK, shell scripts, make)

7 Hypervisors and Virtual Systems (4 hours)

- 7.1 Hypervisors: Type 1 and type 2
- 7.2 Virtual machines: Creating virtual machine in Qemu/Virtual box/VMWare
- 7.3 Container virtualization: Docker and Kubernetes
- 7.4 Power shell and windows subsystem for LINUX (WSL)
- 7.5 Performance optimization and security in virtualized environments

8 Overview of Contemporary OS (3 hours)

- 8.1 Windows and Linux-based OS
- 8.2 Embedded and mobile OS
- 8.3 IoT and RT operating system
- 8.4 Robot and smart card operating system

Tutorial**(15 hours)**

1. Unix shell programs to do a variety of tasks
2. Problems on scheduling algorithms
3. Problems on page replacement algorithms
4. Problems on deadlock
5. Problems on segmentation and paging
6. Comprehensive study on storage technologies with cost/bit and speed comparisons
7. Study on process management/memory management/IO management/security/file system of a selected OS

Practical**(22.5 hours)**

1. Basic Unix commands
2. Shell Programming
3. Implementation of the ls and grep commands
4. Programs using the I/O system calls of the UNIX operating system
5. Implementation of scheduling algorithms and the producer-consumer problem using semaphores
6. Implementation of some memory management schemes such as paging and segmentation
7. Term Project

Final Exam

The questions will cover all the chapters in the syllabus. The evaluation scheme will be as indicated in the table below:

Chapter	Hours	Marks Distribution*
1	6	8
2	7	10
3	10	13
4	9	12
5	3	4
6	3	4
7	4	5
8	3	4
Total	45	60

* There may be minor deviation in marks distribution.

Reference

1. Tanenbaum, A.S., Bos, H. (2024). Modern Operating Systems .3rd Edition. PHI.
2. Stalling, W. (2008). Operating Systems. 6th Edition. Pearson Education.
3. Chaturvedi, A., Rai, B.L. (2017). UNIX and Shell Programming. University Science Press.



YEAR III, PART I

Semester 5

6 COURSES IN THIS SEMESTER

ENSH 304	Probability and Statistics	98
ENCT 301	Database Management System	101
ENCT 302	Web Application Programming	105
ENCT 303	Computer Organization and Architecture	109
ENCT 304	Computer Networks	113
ENCT 325-344	Elective I	117

5

PROBABILITY AND STATISTICS

ENSH 304

Lecture : 3
Tutorial : 1
Practical : 0

Year : III
Part : I

Course Objectives:

The objective of this course is to equip students with foundational knowledge in probability and statistics, focusing on core concepts essential for engineering applications. Students will develop essential skills in statistical data analysis, enabling them to apply various statistical techniques to address real-world engineering challenges. Additionally, the course emphasizes the interpretation and effective communication of statistical results, preparing students to make informed, data-driven decisions in their professional practice.

- 1 Descriptive Statistics and Basic Probability (6 hours)**
 - 1.1 Introduction to statistics and its importance in engineering
 - 1.2 Measure of central tendency and measure of variation
 - 1.3 Graphical representation of data: Histograms, box plots and scatter plots
 - 1.4 Basic probability concepts, additive law, multiplicative law
 - 1.5 Conditional probability and Bayes' theorem
- 2 Probability Distributions and Sampling Distribution (14 hours)**
 - 2.1 Random variables: Discrete and continuous
 - 2.2 Expectation and variance of discrete and continuous random variables
 - 2.3 Discrete probability distributions: Binomial, Poisson, negative Binomial
 - 2.4 Continuous probability distributions: Normal, Gamma, Chi-Square
 - 2.5 Population and sample
 - 2.6 Sampling distribution of mean and proportion
 - 2.7 Central limit theorem
- 3 Statistical Inference (14 hours)**
 - 3.1 Point estimations and properties of estimators
 - 3.2 Confidence intervals for mean and proportions
 - 3.3 Hypothesis testing, parametric and non-parametric tests, procedure of hypothesis
 - 3.4 Hypothesis testing of mean (Single mean, two means, paired t-test and one-way)
 - 3.5 Goodness of fit tests and independence of attributes (Chi-square and Kolmogorov–Smirnov test)

4 Correlation and Regression (6 hours)

- 4.1 Correlation analysis and test of linear correlation
- 4.2 Simple regression analysis, the concept of explained, unexplained, and total
- 4.3 Multiple regression analysis

5 Statistical Quality Control (5 hours)

- 5.1 Quality control and its importance in engineering
- 5.2 Control charts for variables (X-bar, R-chart, P-chart)
- 5.3 Six sigma concepts

Tutorial (15 hours)

1. Visualize data, compute central tendency, and variance in engineering problems using computer software
2. Solve different engineering problems involving probability
3. Solve different engineering problems involving discrete probability distribution and its interpretation
4. Solve different engineering problems involving continuous probability distribution and its interpretation
5. Analyze numerical engineering datasets, perform normality tests, confidence intervals, significance tests of means, and ANOVA
6. Analyze categorical engineering datasets, perform crosstabulation, proportion tests, Chi-Square tests, and draw conclusions using computer software
7. Calculate the correlation coefficient and perform correlation tests on engineering data
8. Fit and interpret simple/multiple regression models on engineering data using computer software
9. Use control charts for process monitoring on sample engineering data.
10. Create control charts using computer software

Final Exam

The questions will cover all the chapters in the syllabus. The evaluation scheme will be as indicated in the table below:

Chapter	Hours	Marks distribution*
1	6	10
2	14	15
3	14	20
4	6	10
5	5	5
Total	45	60

* There may be minor deviation in marks distribution.

References

1. Ronald, E.W., Raymond, H.M., Sharon, L.M. (2012). Probability & Statistics for Engineers & Scientists (9th edition). Boston USA: Prentice Hall.
2. Richard A.J. (2018). Probability and Statistics for Engineers (9th edition). Edinburgh Gate: Pearson Education Limited
3. Sheldon M.R. (2009). Introduction to Probability and Statistics for Engineers and Scientists (4th edition). London: Elsevier Inc.
4. Jay L.D. (2012). Probability and Statistics for Engineering and Sciences. Boston: Thomson Brooks/Cole
5. Brian S.E., Ibrsten H. (2010). A Handbook of Statistical Analyses Using R (2nd edition). London: CRC Press Taylor & Francis Group
6. Andy F. (2018). Discovering Statistics Using IBM SPSS Statistics (5th edition). London: SAGE Publications.

DATABASE MANAGEMENT SYSTEM

ENCT 301

Lecture : 3

Tutorial : 1

Practical : 3

Year : III

Part : I

Course Objectives:

The objective of this course is to provide a comprehensive understanding of the principles and practices involved in the design and implementation of database systems. It enables students to develop data models, and perform data modification, processing, and management efficiently. The course also introduces advanced concepts such as object-oriented databases, data warehousing, data processing, and big data management.

1 Introduction (3 hours)

- 1.1 Application and evolution of database
- 1.2 Data abstraction (Physical, logical, and view level) and data independence
- 1.3 Schema and instances

2 Data Models (7 hours)

- 2.1 Introduction to data models (Entity-relationship, relational, object model, hierarchical, network, graph data models)
- 2.2 E-R model
 - 2.2.1 Entities and entity sets
 - 2.2.2 Attributes and keys
 - 2.2.3 Strong and weak entity sets
 - 2.2.4 Relationship and relationship sets (Mapping cardinalities)
 - 2.2.5 Specialization, generalization and aggregation
- 2.3 Relational model
 - 2.3.1 Concept of relational model, key constraints
 - 2.3.2 Converting ER model into relational model

3 Relational Query Languages (7 hours)

- 3.1 Relational algebra
- 3.2 Concept of DDL, DML and DCL
- 3.3 Overview of the SQL query language-DDL and DML queries
- 3.4 Set operations
- 3.5 Aggregate functions – GROUP BY – HAVING

- 3.6 Joins and types of joins
- 3.7 Nested sub queries
- 3.8 Database modification (Insert, update, delete)
- 3.9 Views
- 3.10 Triggers and stored procedures
- 3.11 Privilege and roles management – GRANT and REVOKE statements

4 Database Constraints and Normalization (6 hours)

- 4.1 Integrity constraints and domain constraints
- 4.2 Assertions
- 4.3 Functional dependencies
- 4.4 Different normal forms (1NF, 2NF, 3NF, BCNF)

5 Query Processing and Optimization (4 hours)

- 5.1 Query processing, optimization and evaluation
- 5.2 Transformation of relational expressions
- 5.3 Techniques of implementing query optimization - Cost based optimization and heuristic optimization
- 5.4 Query evaluation -Materialization and pipelining
- 5.5 Denormalization for performance
- 5.6 Materialized view
- 5.7 Performance tuning

6 File Structure and Hashing (5 hours)

- 6.1 Disks and storage
- 6.2 Records organizations
- 6.3 Ordered indices
- 6.4 B+ tree index
- 6.5 Hashing concepts - Static and dynamic hashing

7 Transaction Processing and Concurrency Control (5 hours)

- 7.1 Transaction and transaction model - State diagram
- 7.2 Acid properties
- 7.3 Concurrent execution of transactions
- 7.4 Serializability (Conflict and view serializability)
- 7.5 Lock based protocols
- 7.6 Deadlock handling and prevention
- 7.7 Multiple granularity

8 Crash Recovery (4 hours)

- 8.1 Failure classification
- 8.2 Recovery and atomicity

- 8.3 Log-based recovery
- 8.4 Shadow paging
- 8.5 High availability using remote backup systems

9 Advanced Database Concepts

(4 hours)

- 9.1 Concept of object-oriented databases
- 9.2 Distributed database model
- 9.3 Concept of data warehousing and online analytical processing
- 9.4 Basic concepts of NoSQL and big data

Tutorial

(15 hours)

- 1. Designing ER model for sample cases
- 2. Converting an ER diagram to a relational schema
- 3. Practice with simple relational algebra queries
- 4. Insert, update and delete with relational algebra queries using assignment operator
- 5. Executing DDL commands to create a database, define primary and foreign key constraints
- 6. Simple queries with SELECT and filtering
- 7. Complex queries with joins and sub-queries
- 8. Aggregating data with GROUP BY and HAVING
- 9. Database manipulation with INSERT, UPDATE and DELETE queries
- 10. Creating and using views
- 11. Creating users, roles and using GRANT and REVOKE statements
- 12. Triggers and stored procedures
- 13. Identifying functional dependencies and normal forms
- 14. Decomposing a table into BCNF and 3NF
- 15. Introduction to NoSQL databases with MongoDB

Practical

(45 hours)

- 1. Database server installation and configuration
- 2. DB client installation and connection to DB server. Introduction and practice with SELECT command with the existing DB
- 3. Further practice with DML queries – Select, insert, update and delete
- 4. Advanced queries with joins and subqueries
- 5. Aggregation and grouping
- 6. Practice with DDL commands – Create/alter/drop table, integrity constraints and views
- 7. Triggers and stored procedures
- 8. Query processing, optimization, performance tuning and database administration
- 9. Group project work

Final Exam

The questions will cover all the chapters in the syllabus. The evaluation scheme will be as indicated in the table below:

Chapter	Hours	Marks distribution*
1	3	3
2	7	9
3	7	9
4	6	9
5	4	6
6	5	8
7	5	8
8	4	4
9	4	4
Total	45	60

* There may be minor deviation in marks distribution.

References

1. Silberschatz, A., Korth, H.F., Sudarshan, S. (2019). Database system concepts. McGraw-Hill.
2. Elmasri, R., Navathe, S.B. (2021). Fundamentals of database systems. Pearson.
3. Ramakrishnan, R., Gehrke, J. (2002). Database management systems. McGraw-Hill.
4. Connolly, T.M., Begg, C.E. (2021). Database systems: A practical approach to design, implementation, and management. Pearson.

WEB APPLICATION PROGRAMMING

ENCT 302

Lecture : 3
Tutorial : 1
Practical : 3

Year : III
Part : I

Course Objectives:

The objective of this course is to provide comprehensive understanding of web technologies for developing responsive and interactive web applications. The course emphasizes the use of modern front-end frameworks and tools, as well as techniques for integrating front-end interfaces with back-end APIs and databases. By the end of the course, students will be able to design, implement and evaluate contemporary applications while exploring current trends and future directions in web development.

1 Introduction (6 hours)

- 1.1 Overview of web applications and evolution of web architecture
- 1.2 Client-server architecture, HTTP, HTTPS, URLs, DNS, web browsers
- 1.3 Html basics: Syntax, tags, attributes, forms and inputs, tables, lists, multimedia, semantic HTML5 elements
- 1.4 CSS basics: Selectors, properties, values, box model
- 1.5 CSS framework: Bootstrap

2 JavaScript and Client-Side Programming (12 hours)

- 2.1 JavaScript essentials
 - 2.1.1 Data types, variables, control structures, functions
 - 2.1.2 Dom manipulation, events, and form validation
 - 2.1.3 Local storage and session storage
 - 2.1.4 GUI interactions
 - 2.1.5 JavaScript library: jQuery
- 2.2 Modern JavaScript (ES6+)
 - 2.2.1 Arrow functions, destructuring, spread/rest operators
 - 2.2.2 Callbacks, promises, async/await
 - 2.2.3 Modules and imports
- 2.3 Client-side applications
 - 2.3.1 Client-side web application development using React JS library
 - 2.3.2 Traditional multi-page apps vs. single-page apps (SPAs)
 - 2.3.3 Component-based UI and props
 - 2.3.4 State management and data flow
 - 2.3.5 Client-side routing and navigation without reloads
 - 2.3.6 Fetch API: Fetching, displaying data, handling errors and loading states

2.3.7 Comparison of modern JavaScript libraries: React, Angular, Vue

3 Server-Side Web Programming (9 hours)

- 3.1 MVC architecture in web development
- 3.2 Role of backend in web applications
- 3.3 Backend web framework: Django
- 3.4 Handling requests and responses
- 3.5 Form data handling and sessions
- 3.6 Routing, middleware, templating concepts
- 3.7 Overview and comparison of backend frameworks: Django, flask, fastapi, dot NET MVC framework, ruby on rails, java spring boot, node.js
- 3.8 Database integration: Relational versus NoSQL, CRUD operations, ORM concept
- 3.9 Authentication and authorization: Cookies, sessions, JWT
- 3.10 Middleware for logging, error handling, and security

4 Web Services and APIs (7 hours)

- 4.1 API basics: Role in web applications
- 4.2 REST principles and design, RESTful APIs
- 4.3 JSON versus XML data exchange
- 4.4 Data validation and serialization
- 4.5 Microservices

5 Web Application Security (6 hours)

- 5.1 Common vulnerabilities: XSS, SQL injection, CSRF
- 5.2 Security best practices: Input validation, sanitization, https, secure cookies, env variables
- 5.3 Authentication practices and token handling
- 5.4 Security in full-stack apps: CORS, safe sessions

6 Web Application Deployment and Modern Trends (5 hours)

- 6.1 Full-stack development
- 6.2 Testing and QA
- 6.3 DevOps, continuous integration (CI) and continuous delivery (CD)
- 6.4 Progressive web apps (PWAs), responsive design and usability

Tutorial (15 hours)

- 1. Walkthrough of client-server request flow (DNS→HTTP/HTTPS→ Browser)
- 2. Guided coding: HTML page with semantic tags, styled with CSS grid/flexbox
- 3. JavaScript and client-side programming
- 4. Hands-on exercises with DOM manipulation, event handling
- 5. Form validation and local storage demo

6. Guided task: Small SPA with component-based UI and fetch API integration
7. Discuss request/response cycle with server framework
8. Guided coding: Simple CRUD using Django/Flask (e.g., library system)
9. Session handling and authentication example
10. REST principles with examples
11. Hands-on exercise: Build a simple REST API endpoint and test with fetch/postman
12. Demonstration of XSS and SQL Injection vulnerabilities
13. Guided exercise: Secure login form with input validation and session/token handling
14. Discussion on PWAs, serverless, microservices
15. Case study/tutorial: Compare a traditional app vs. PWA features

Practical

(45 hours)

1. Build a responsive personal portfolio page using semantic HTML5, CSS grid/flexbox, and media queries
2. Create a dynamic to-do list app with DOM events, localStorage, and JavaScript event handling (using jQuery)
3. Develop a CRUD application (e.g., library system) using a server-side framework (Django/Flask/FastAPI)
4. Create a frontend in React that fetches data from your backend API and displays it dynamically.
5. Extend the CRUD app with user authentication (Login/logout), form validation, and secure token/session handling
6. Develop a comprehensive project that encompasses the complete process of front-end and back-end web application development and deployment. The project should be carried out over at least six to seven lab sessions, covering the following stages: Project proposal, front-end UI/UX design, back-end and database design, API design and integration, testing and deployment, and culminating in a final presentation and demonstration

Final Exam

The questions will cover all the chapters in the syllabus. The evaluation scheme will be as indicated in the table below:

Chapter	Hours	Marks distribution*
1	6	8
2	12	16
3	9	12
4	7	9
5	6	8
6	5	7
Total	45	60

* There may be minor deviation in marks distribution.

References

1. Copes, F. (2019). The JavaScript Handbook. (<https://flaviocopes.com/the-javascript-handbook-2019-edition/>)
2. Nixon, R. (2025). Learning PHP, MySQL & JavaScript: A Step-by-Step Guide to Creating Dynamic Websites. O'Reilly Media.
3. Vincent, W. S. (2025). Django for Beginners: Build websites with Python and Django.
4. Richardson, L., & Amundsen, M. (2013). RESTful Web APIs: Services for a Changing World. O'Reilly Media.
5. Hoffman, A. (2024). Web Application Security: Exploitation and Countermeasures for Modern Web Applications (2nd ed.). O'Reilly Media.

COMPUTER ORGANIZATION AND ARCHITECTURE

ENCT 303

Lecture : 3
Tutorial : 1
Practical : 3/2

Year : III
Part : I

Course Objectives:

The objective of this course is to provide the organization and architectural concept of computer system including processor architecture, computer arithmetic, memory system, I/O organization, multiprocessor and multicore.

1 Introduction (5 hours)

- 1.1 Organization and architecture
- 1.2 Structure of a computer, single processor, multi-core computer
- 1.3 Performance assessment
 - 1.3.1 Clock speed and instruction per second
 - 1.3.2 Instruction execution rate: CPI, MIPS rate, MFLOPS rate, arithmetic mean, harmonic mean, speed metric, geometric mean, rate metric, Amdahl's law, speed up
- 1.4 Computer function
 - 1.4.1 Instruction fetch and execute
 - 1.4.2 Instruction cycle state diagram
- 1.5 Computer component, interconnection structure, bus interconnection, PCI
- 1.6 RISC architecture, overlapped register windows, Berkeley RISC

2 Central Processing Unit (CPU) (7 hours)

- 2.1 Processor bus organization
- 2.2 Processor register organization: Control word, examples of microoperations
- 2.3 Stack organization: Register stack, memory stack, reverse polish notation, evaluation of arithmetic expressions
- 2.4 Instruction formats: CPU organization, zero and more address instruction formats
- 2.5 Addressing modes: Types, examples, strengths and weaknesses
- 2.6 Instruction set
 - 2.6.1 Data transfer instruction
 - 2.6.2 Data manipulation instruction: Arithmetic, logical and shift operations
 - 2.6.3 Program control instruction
- 2.7 Status bit conditions
- 2.8 Interrupt: Definition, types, processing and ISR

- 3 Control Unit (5 hours)**
- 3.1 Hardwired control unit
 - 3.2 Microprogrammed control unit
 - 3.3 Microinstructions, control memory organization, Wilkes control
 - 3.4 Microinstruction sequencing: Design considerations, sequencing techniques, address generation, microinstruction encoding
 - 3.5 Application of microprogramming
 - 3.6 Microinstruction execution
- 4 Memory System (7 hours)**
- 4.1 Characteristics of memory system
 - 4.2 Memory classification and hierarchy
 - 4.3 Semiconductor memory and its types, read only memory, read/write memory
 - 4.4 Ram modules and interfaces: DDR, DIMM and SODIMM
 - 4.5 Cache memory
 - 4.5.1 Cache principles
 - 4.5.2 Elements of cache design: Cache size, mapping function, replacement algorithms, write policy, block size, single and multi-level caches and unified versus split cache
 - 4.6 External memory
 - 4.6.1 Magnetic disk
 - 4.6.2 RAID: Level 1 to 5
 - 4.6.3 Optical memory
 - 4.6.4 Magnetic tape
 - 4.6.5 SSD
- 5 Computer Arithmetic (8 hours)**
- 5.1 ALU (Arithmetic and logic unit)
 - 5.2 Integer representation: Sign-magnitude representation, two's complement representation, converting between different bit lengths, fixed-point representation
 - 5.3 Integer arithmetic
 - 5.3.1 Addition and subtraction algorithm
 - 5.3.2 Multiplication algorithm
 - 5.3.3 Division algorithm
 - 5.4 Floating-point arithmetic
 - 5.4.1 Floating-point representation: Principles, IEEE standard for binary floating-point arithmetic algorithm
 - 5.4.2 Addition and subtraction algorithm
 - 5.4.3 Multiplication algorithm
 - 5.4.4 Division algorithm

- 6 Pipelining and Vector Processing** **(4 hours)**
- 6.1 Pipelining and its importance
 - 6.2 Instruction and arithmetic pipelining
 - 6.3 Pipelining hazards: Data, structural and control hazards
 - 6.4 RISC pipeline
 - 6.5 Parallel processing
 - 6.6 Vector processing
 - 6.6.1 Vector operations
 - 6.6.2 Matrix multiplication
 - 6.6.3 Memory interleaving
 - 6.6.4 Superscalar processors
 - 6.6.5 Supercomputers
 - 6.7 Array processors: Attached array processor and SIMD array processor
- 7 Input/Output** **(5 hours)**
- 7.1 External devices
 - 7.2 I/O modules: Module function, module structure
 - 7.3 Programmed I/O, I/O commands, I/O instructions, flowchart
 - 7.4 Interrupt driven I/O, interrupt processing and flowchart
 - 7.5 Direct memory access (DMA): Drawbacks of programmed and interrupt-driven I/O, DMA function, typical DMA block diagram and possible DMA configuration
 - 7.6 I/O channels and processors: The evolution of the I/O function, characteristics of I/O channels
 - 7.7 The external interface: Types of interfaces, point-to point and multiple configurations, small computer system interface (SCSI)
- 8 Multiprocessor System** **(4 hours)**
- 8.1 Multiprocessor computers and their characteristics
 - 8.2 Multi-core computers and their architecture
 - 8.3 Interconnection structure: Time-shared common bus, multiport memory, crossbar switch, multistage switching network and hypercube system
 - 8.4 Interprocessor arbitration
 - 8.5 Interprocessor communication and synchronization
- Tutorial** **(15 hours)**
- 1. Numerical examples on performance assessment
 - 2. Design of arithmetic circuit, logic circuit and ALU
 - 3. Coding examples covering different instruction formats
 - 4. Microprogramming examples in CU
 - 5. Cache memory mapping: Hit and miss ratio
 - 6. Numerical examples for various arithmetic algorithms
 - 7. Memory and I/O interfacing

Practical

(22.5 hours)

There will be about 6 lab exercises based on various arithmetic algorithm, cache memory and vector processing using any high-level language, MATLAB or other simulator.

1. Addition and subtraction algorithm
2. Multiplication algorithm
3. Division algorithm
4. Cache mapping techniques
5. ALU implementation
6. Vector processing implementation

Final Exam

The questions will cover all the chapters in the syllabus. The evaluation scheme will be as indicated in the table below:

Chapter	Hours	Marks distribution*
1	5	6
2	7	10
3	5	6
4	7	10
5	8	10
6	4	6
7	5	6
8	4	6
Total	45	60

* There may be minor deviation in marks distribution.

References

1. Stallings, W. (2018). Computer organization and architecture. Prentice Hall of India.
2. Mano, M. M. (2008). Computer system architecture. Pearson Education.
3. Hennessy, J. L., Patterson, D. A. (2000). Computer architecture: A quantitative approach. Harcourt Asia.

COMPUTER NETWORKS

ENCT 304

Lecture : 3
Tutorial : 1
Practical : 3

Year : III
Part : I

Course Objectives:

The objective of this course is to provide fundamental concepts and principles of computer networks. It focuses on various network architectures, protocols, standards, and the networking device functionalities. The course also aims to equip students with the skills to design, implement, test and troubleshoot basic network configurations.

1 Introduction (5 hours)

- 1.1 Computer network, importance and applications
- 1.2 Types of computer networks
- 1.3 Network topologies
- 1.4 Client/server and P2P networks
- 1.5 Protocols and standards, need of standardization
- 1.6 Layered network architecture, OSI and TCP/IP reference models
- 1.7 Addressing at different layers
- 1.8 Networking devices: Repeater, hub, bridge, switch, router, gateway

2 Physical Layer (5 hours)

- 2.1 Network monitoring: Bandwidth, throughput, delay, round-trip-time
- 2.2 Multiplexing - FDM, TDM, WDM
- 2.3 Switching - circuit switching, packet switching, datagram and virtual circuit switching
- 2.4 Transmission media - Guided (twisted pair, coaxial, optical fiber) and unguided media (radio waves, microwaves, infrared)
- 2.5 Ethernet cable standards (twisted pair and optical fiber)
- 2.6 Data encoding: Manchester, NRZ-I, MLT3, 4B/5B

3 Data Link Layer (8 hours)

- 3.1 Data link layer functions - Framing, error control, flow control, access control
- 3.2 Error control codes: Parity, checksum, CRC
- 3.3 Data link protocols: PPP, HDLC
- 3.4 Logical link control (LLC) and media access control (MAC) sublayers
- 3.5 Random access protocols: ALOHA, CSMA, CSMA/CD, CSMA/CA

- 3.6 Token based protocols: Token bus, token ring, FDDI
- 3.7 Ethernet (IEEE 802.3) and its evolution, bridged and switched Ethernet
- 3.8 WLAN (Wi-Fi) and IEEE 802.11
- 3.9 VLAN and its importance
- 3.10 Virtual circuit switching: ATM, MPLS

4 Network Layer (12 hours)

- 4.1 Network layer functions and services
- 4.2 Connection oriented and connectionless network services
- 4.3 Logical addressing and its importance
- 4.4 IPv4 addressing and classes, private and public IP address, sub-netting and super-netting, VLSM, CIDR
- 4.5 Unicast, multicast and broadcast addresses
- 4.6 Routing and types of routing: Static and dynamic, unicast and multicast, interior and exterior, distance vector and link state
- 4.7 Routing protocols: RIP, OSPF, EIGRP, BGP
- 4.8 Network address translation (NAT)
- 4.9 IPv6 addressing, need and features of IPv6
- 4.10 Protocols: IPv4, ARP, ICMP, IPv6, ICMPv6
- 4.11 Transition from IPv4 to IPv6 and strategies
- 4.12 Traffic shaping and congestion control mechanisms

5 Transport Layer (5 hours)

- 5.1 Transport layer functions and services
- 5.2 Elements of transport layer protocols: process-to-process communication, addressing, multiplexing and de-multiplexing, segmentation and reassembly, error control, flow control
- 5.3 Transport protocols: TCP and UDP
- 5.4 TCP services: connection setup and release, reliable, stream oriented, flow-control, error-control
- 5.5 Introduction to socket and socket programming

6 Upper Layers and Network Design (6 hours)

- 6.1 Functions and services of session, presentation and application layers
- 6.2 Introduction to upper layer protocols: DHCP, HTTP, HTTPS, FTP, SMTP, POP, IMAP
- 6.3 DNS, its function and DNS queries
- 6.4 Network management tools and protocols (SNMP)
- 6.5 Basics of web, e-mail, DNS and proxy server configurations
- 6.6 VoIP, FoIP and IP interconnection
- 6.7 Network design and configuration guidelines

7 Advanced Topics

(4 hours)

- 7.1 Introduction to software-defined networking (SDN), basic architecture, importance and applications
- 7.2 Network security and its importance
- 7.3 Overview of content delivery networks (CDNs), named data networking (NDN), intent based networking (IBN)
- 7.4 Quantum network, network virtualization and recent trends in networking

Tutorial

(15 hours)

1. Comparison of OSI and TCP/IP models
2. Comparison of twisted-pair, coaxial, and fiber optic in terms of cost, bandwidth, applications and so on
3. Datalink layer frames and framing techniques
4. Comparison of CSMA/CD and CSMA/CA
5. IP address classes, reserved IP addresses, subnetting (FLSM and VLSM), supernetting, comparison between IPv4 and IPv6
6. Comparison of different routing protocols with their applications
7. TCP and UDP protocols, their applications
8. TCP 3-way handshake with message flow
9. Application protocols to services (HTTP → Web, SMTP → Email, FTP → File transfer, DNS → Name resolution)
10. Working of DNS

Practical

(45 hours)

1. Network cabling: Preparation of network cables and their uses
2. Network commands which are very important for testing and troubleshooting
3. Basic router configurations, basic network setup: IP address, subnet mask, default gateway
4. Subnetting and supernetting
5. Configuration of static routes and default routes
6. Dynamic routing configurations: RIP, EIGRP, OSPF, BGP
7. Configuration of switch, VLAN configuration and inter-VLAN routing
8. Server configurations: DHCP, web, DNS
9. Wireless network setup and packet analysis
10. Network troubleshooting: Diagnosing and resolving network issues
11. Network design: Design a complete network on given requirements using simulators like Cisco packet tracer or GNS3 or Mininet or equivalent tool
12. Presentation and review

Final Exam

The questions will cover all the chapters in the syllabus. The evaluation scheme will be as indicated in the table below:

Chapter	Hours	Marks distribution*
1	5	6
2	5	6
3	8	12
4	12	15
5	5	6
6	6	9
7	4	6
Total	45	60

* There may be minor deviations in marks distribution.

References

1. Forouzan, B. A. (2012). Data communications and networking. McGraw-Hill.
2. Kurose, J.F., Ross, K.W. (2017). Computer networking: A top-down approach. Pearson.
3. Tanenbaum, A.S., Feamster, N. (2021). Computer networks. Pearson.
4. Stevens, W.R., Fall, K.R. (2011). TCP/IP illustrated: The protocols. Addison-Wesley.
5. RFCs and online resources for protocol specifications.

ADVANCED PYTHON PROGRAMMING FOR DATA SCIENCE

ENCT 325

Lecture : 3
Tutorial : 2
Practical : 1

Year : III
Part : I

Course Objectives:

The objective of this course is to develop advanced proficiency in Python programming for data science applications. It focuses on efficient coding practices, sophisticated data manipulation, statistical analysis, and data visualization using modern Python libraries. Students will also learn fundamental data engineering and pipeline design concepts to automate and scale real-world data workflows.

1 Advanced Python Concepts and Best Practices (7 hours)

- 1.1 Review of Python essentials and coding conventions
- 1.2 Advanced data structures: Collections, iterators, generators, and decorators
- 1.3 Functions and lambda expressions
- 1.4 Object-Oriented Programming for data science applications
- 1.5 Exception handling, debugging, and logging
- 1.6 Working with modules and packages

2 Data Sources and APIs (7 hours)

- 2.1 Reading and writing structured/unstructured data (CSV, JSON, Excel, text)
- 2.2 Database access with relational database and non-relational database
- 2.3 Accessing and processing data from APIs (REST, SOAP)
- 2.4 Web scraping using requests and BeautifulSoup
- 2.5 Handling large datasets with chunking and lazy evaluation

3 Advanced Data Wrangling and Transformation (9 hours)

- 3.1 Advanced Pandas operations: Merging, joining, reshaping, pivoting
- 3.2 Handling missing, categorical, and time-series data
- 3.3 Feature transformation, scaling, and encoding
- 3.4 Memory optimization and efficient data processing
- 3.5 Building a reusable data-cleaning pipeline
- 3.6 Introduction to data pipeline components (Ingestion, transformation, storage)

4 Applied Statistics and Exploratory Analysis (7 hours)

- 4.1 Statistical measures: Correlation, covariance, skewness, kurtosis
- 4.2 Probability review, sampling, and hypothesis testing
- 4.3 Regression and trend analysis using stats models
- 4.4 Exploratory data analysis (EDA) using descriptive and inferential methods
- 4.5 Automation of EDA workflows using Python

5 Data Visualization and Storytelling (7 hours)

- 5.1 Principles of effective visualization and dashboard design
- 5.2 Visualization with Matplotlib: Line, bar, histogram, scatter, subplots
- 5.3 Seaborn for statistical visualization: Box plot, pair plot, heat map
- 5.4 Interactive visualization using Plotly
- 5.5 Visualization driven insight generation
- 5.6 Case study: End-to-end visualization and reporting project

6 Data Engineering and Automation (8 hours)

- 6.1 Overview of data engineering in applied data science
- 6.2 Designing and implementing ETL pipelines
- 6.3 Automating workflows with schedulers (CRON, schedule)
- 6.4 Logging, monitoring, and error handling in pipelines
- 6.5 Data storage and retrieval strategies for pipelines
- 6.6 Automated report generation (Excel, HTML, PDF)
- 6.7 Case study: End-to-end automated analytics pipeline

Tutorial (30 hours)

- 1. Python refresher and best practices: Review of Python syntax, indentation, and PEP8 coding conventions through short exercises
- 2. Iterators, generators, and decorators: Writing small programs using generators for data streaming and decorators for function modification
- 3. Object-oriented programming (OOP): Designing simple class-based programs and demonstrating inheritance and encapsulation in python
- 4. Data access and integration: Reading data from CSV, Excel, JSON, and APIs; Discussion on best practices for data ingestion
- 5. Database and SQL interaction: practice using SQLite and SQLAlchemy to query and manipulate structured datasets
- 6. Web scraping practice: Extracting tabular data using requests and BeautifulSoup; Handling exceptions and encoding issues
- 7. Advanced Pandas operations: Hands-on merging, reshaping, pivoting, and group-by operations for complex data manipulation
- 8. Data cleaning and transformation: Exercises on handling missing values, encoding categorical data, and normalization techniques
- 9. Statistical computation and EDA: Performing descriptive analysis, correlation, and hypothesis testing using Python libraries

10. Data visualization practice: Creating comparative plots using matplotlib, Seaborn, and Plotly; Customizing themes and layouts
11. Pipeline and automation concepts: Designing pseudocode and flow diagrams for ETL data pipelines and discussing error handling strategies
12. Mini case study discussion: Guided review of a small end-to-end applied data pipeline from ingestion to visualization and reporting

Practical

(15 hours)

1. Setting up Python environment for applied data workflows and writing modular programs using OOP and functions
2. Collecting data via APIs and web scraping
3. Building advanced data cleaning and transformation pipelines using Pandas
4. Conducting exploratory data analysis and statistical summaries
5. Developing interactive visualizations using Plotly, matplotlib and Seaborn
6. Automating ETL tasks and data refresh using Python schedulers
7. Generating summary dashboards and automated analytical reports
8. Mini Project: Build a complete applied data pipeline from ingestion to visualization and reporting on a real-world dataset

Final Exam

The questions will cover all the chapters in the syllabus. The evaluation scheme will be as indicated in the table below:

Chapter	Hours	Marks distribution*
1	7	9
2	7	9
3	9	12
4	7	10
5	7	9
6	8	11
Total	45	60

* There may be minor deviation in marks distribution.

References

1. McKinney, W. (2022). Python for data analysis. O'Reilly Media.
2. VanderPlas, J. (2016). Python data science handbook. O'Reilly Media.
3. Beazley, D. (2021). Python cookbook. O'Reilly Media.
4. Grus, J. (2022). Data science from scratch. O'Reilly Media.

COMPILER DESIGN

ENCT 326

Lecture : 3
Tutorial : 2
Practical : 1

Year : III
Part : I

Course Objectives:

The objective of this course is to develop a clear understanding of the structure, components, and design principles of a compiler. It focuses on the key phases of compilation, including lexical analysis, parsing, semantic analysis, and code generation. The course also introduces optimization techniques and error-handling strategies to enhance the efficiency and reliability of compilers.

1 Introduction (5 hours)

- 1.1 Compiler and Interpreter
- 1.2 Compiler structure: Analysis and synthesis model of compilation, different sub-phases within the analysis and synthesis phases
- 1.3 Basic concepts like preprocessor, macros, symbol table, and error handler

2 Lexical Analysis (9 hours)

- 2.1 The role of the lexical analyzer
- 2.2 Specification and recognition of tokens
- 2.3 Input buffering
- 2.4 Finite automata and their relevance to compiler construction
- 2.5 Design of a lexical analyzer generator
- 2.6 Error handling in lexical analysis

3 Syntax Analysis (Parsing) (8 hours)

- 3.1 Role of syntax analysis
- 3.2 Context-free grammars and parse trees
- 3.3 Parsing techniques: Top-down (recursive descent, LL(1)) and bottom-up (LR(1))
- 3.4 Operator precedence
- 3.5 Handling issues like left recursion and ambiguity

4 Syntax-Directed Translation and Semantic Analysis (9 hours)

- 4.1 Syntax-directed definitions (SDD) and syntax-directed translation (SDT)
- 4.2 Synthesized and inherited attributes
- 4.3 Syntax tree construction

- 4.4 Intermediate code generation
- 4.5 Semantic analysis, including static and dynamic checks
- 4.6 Type checking and type systems
- 4.7 Symbol table construction, attributes, and data structures for symbol table
- 4.8 Run-time storage management

5 Code Generation (7 hours)

- 5.1 Factors affecting a code generator
- 5.2 Basic blocks and flow graphs
- 5.3 Optimization of basic blocks
- 5.4 Register allocation and assignment
- 5.5 Dynamic programming code-generation algorithm

6 Code Optimization (7 hours)

- 6.1 Loops in flow graphs, constant propagation, partial redundancy elimination
- 6.2 Data-flow analysis and solving data-flow equations
- 6.3 Storage organization and allocation strategies (Stack, heap)
- 6.4 Need and criteria of code optimization and basic optimization techniques
- 6.5 Basic block optimization
- 6.6 Case studies of some compilers, like the c compiler, the C++ compiler

Tutorial (30 hours)

- 1. Description of various phases and structures of a compiler.
- 2. Understanding of regular expressions
- 3. DFA exercises
- 4. NFA, Thomson's construction
- 5. Bottom-up parsing
- 6. Top down parsing
- 7. Building LR(1) and LL(1) tables
- 8. Creating graphical intermediate code generations
- 9. Creating linear intermediate code generations
- 10. Building hash tables
- 11. Basic exercise on designing a code generator
- 12. Peephole optimization
- 13. Designing flow graphs
- 14. Data flow analysis
- 15. Solving data-flow equation

Practical (15 hours)

- 1. Setting lexical analysis
- 2. Introduction to Lex/Flex tools
- 3. Implementing context-free grammars
- 4. Designing a basic recursive descent parser, computing first and follow values, predictive parsing table construction and LL (1) grammar

5. Syntax analysis with YACC/Bison
6. Semantic analysis and intermediate code generation

Final Exam

The questions will cover all the chapters in the syllabus. The evaluation scheme will be as indicated in the table below:

Chapter	Hours	Marks distribution*
1	5	7
2	9	12
3	8	11
4	9	12
5	7	9
6	7	9
Total	45	60

* There may be minor deviation in marks distribution.

References

1. Aho, A. V., Lam, M. S., Sethi, R., Ullman, J. D. (2006). Compilers: Principles, techniques, and tools. Pearson.
2. Cooper, K. D., Torczon, L. (2011). Engineering a compiler. Morgan Kaufmann.
3. Hopcroft, J. E., Motwani, R., Ullman, J. D. (2006). Introduction to automata theory, languages, and computation. Addison-Wesley.
4. Muchnick, S. (1997). Advanced compiler design and implementation (Latest Publication). Morgan Kaufmann.
5. Thain, D. (2020). Introduction to compilers and language design. Douglas Thain.

JAVA PROGRAMMING

ENCT 327

Lecture : 3
Tutorial : 2
Practical : 1

Year : III

Part : I

Course Objectives:

The objective of this course is to equip students with comprehensive knowledge and practical skills in Java programming, enabling them to design and develop robust, modern applications. Emphasis is placed on using the Spring Boot framework to build scalable, production-ready enterprise solutions.

1 Java Fundamentals (7 hours)

- 1.1 Overview of Java ecosystem (JDK, JVM, JRE)
- 1.2 Evolution of Java
- 1.3 Modern development tools (Maven, Gradle, IntelliJ, VS Code)
- 1.4 Review of core Java OOP, collections and exception handling
- 1.5 Functional programming: Lambda, expressions and streams API
- 1.6 Introduction to modern features like records and sealed classes

2 Java I/O and Data Handling (6 hours)

- 2.1 Streams and file handling
 - 2.1.1 Byte and character streams
 - 2.1.2 Reading/writing text files
 - 2.1.3 Object streams and serialization
 - 2.1.4 New I/O (NIO.2) file operations
- 2.2 JSON and data Interchange
 - 2.2.1 Handling JSON data with libraries like Jackson or Gson
 - 2.2.2 JSON vs. XML in REST APIs

3 Database and Web Communication (10 hours)

- 3.1 Java database connectivity (JDBC)
 - 3.1.1 JDBC architecture and drivers
 - 3.1.2 Connection, statement, and result set management
 - 3.1.3 Prepared statements for secure and efficient queries
- 3.2 Network programming and rest
 - 3.2.1 Client-server model and basics of socket communication
 - 3.2.2 Consuming restful APIs using the HTTPClient from Java 11+
 - 3.2.3 Overview of microservices architecture

4 Spring Boot and Modern Web Development (16 hours)

- 4.1 Spring Boot fundamentals; Dependency injection and inversion of control; Creating RESTful APIs with Spring Boot
- 4.2 Data Access with Spring
 - 4.2.1 Overview of ORM and JPA
 - 4.2.2 Building a repository layer with spring data JPA
 - 4.2.3 CRUD operations and transactional management
- 4.3 API Development and Security
 - 4.3.1 REST API design principles
 - 4.3.2 Handling exceptions in spring boot
 - 4.3.3 Overview of spring security for authentication and authorization
- 4.4 Deployment and Project Work
 - 4.4.1 Docker and containerization
 - 4.4.2 Packaging and deploying a Spring Boot application

5 Concurrency and Performance (6 hours)

- 5.1 Overview of multithreading
- 5.2 Thread lifecycle and synchronization
- 5.3 Executor framework and thread pools
- 5.4 Completable future and parallel streams
- 5.5 Best practices for thread-safe programming
- 5.6 Performance of multi-threaded program

Tutorial (30 hours)

- 1. Practice Lambda expressions and Stream API operations
- 2. Implement functional interfaces and method references
- 3. Perform file I/O operations and object serialization
- 4. Implement multithreading and synchronization mechanisms
- 5. Develop a Java program for CRUD operations using JDBC
- 6. Create and configure a Spring Boot project using Spring Initializer
- 7. Implement dependency injection using constructor, setter, and field injection
- 8. Develop restful CRUD APIs using Spring Boot and JPA
- 9. Create custom database queries using derived methods and the @query annotation
- 10. Implement centralized exception handling and API error responses
- 11. Configure authentication and authorization using Spring Security
- 12. Consume third-party rest APIs using WebClient or RestTemplate
- 13. Configure environment-specific properties and external database connections
- 14. Containerize and deploy a spring boot application using Docker
- 15. Write and execute unit and integration tests using JUnit 5 and MockMvc
- 16. Project work: Apply learned concepts by building a small, secure, full-stack REST service

Practical**(15 hours)**

1. Programs using Lambda expressions and Streams API
2. Implementation of file I/O operations
3. Development of a simple client-server application
4. Implementation of a Spring Boot application with a data access layer
5. Containerization of a Spring Boot application using Docker

Final Exam

The questions will cover all the chapters in the syllabus. The evaluation scheme will be as indicated in the table below:

Chapter	Hours	Marks distribution*
1	7	8
2	6	8
3	10	14
4	16	22
5	6	8
Total	45	60

* There may be minor deviation in marks distribution.

References

1. Bloch, J. (2017). Effective Java. Addison-Wesley Professional.
2. Schildt, H. (2024). Java: The Complete Reference. McGraw-Hill Education.
3. Goetz, B., Peierls, T., Bloch, J., Bowbeer, D., Holmes, J., Lea, D. (2006). Java Concurrency in Practice. Addison-Wesley Professional.
4. Walls, C. (2016). Spring Boot in Action. Manning Publications.

QUANTUM COMPUTING

ENCT 328

Lecture : 3
Tutorial : 2
Practical : 1

Year : III

Part : I

Course Objectives:

The objective of this course is to introduce the principles, mathematical foundations, and computational models of quantum computing. It emphasizes on algorithmic insights, and offer practical experience with quantum programming environments. By the end of the course, students will be able to represent and manipulate quantum states and gates, design and simulate basic quantum circuits, implement fundamental quantum algorithms and their speedup mechanisms, and explore applications of quantum machine learning (QML) techniques.

1 Fundamentals of Quantum Concepts (6 hours)

- 1.1 Classical versus Quantum computation
- 1.2 Qubits, superposition, measurement
- 1.3 Postulates of quantum mechanics
- 1.4 Dirac notation and Bloch sphere

2 Quantum Gates and Circuits (6 hours)

- 2.1 Quantum gates (X, Y, Z, H, Phase, CNOT)
- 2.2 Unitary operations and reversibility
- 2.3 Multi-qubit systems and tensor products
- 2.4 Circuit design and visualization

3 Entanglement and Quantum Communication (5 hours)

- 3.1 Entangled states and Bell theorem
- 3.2 EPR paradox
- 3.3 Quantum teleportation and superdense coding
- 3.4 Quantum key distribution protocols: BB84, E91

4 Quantum Algorithms (10 hours)

- 4.1 Quantum parallelism
- 4.2 Deutsch and Deutsch–Jozsa algorithms
- 4.3 Bernstein–Vazirani
- 4.4 Grover's algorithm
- 4.5 Shor's algorithm

- 5 Quantum Hardware and NISQ Era (6 hours)**
- 5.1 Physical realization (Ion trap, superconducting, photonic)
 - 5.2 Noise, decoherence, and qubit metrics
 - 5.3 NISQ model and limitations
 - 5.4 Quantum circuit execution on IBM quantum simulators
- 6 Quantum Error and Correction (4 hours)**
- 6.1 Bit-flip and phase-flip errors
 - 6.2 Simple quantum error detection
 - 6.3 Shor's 9-qubit code
 - 6.4 Surface code overview
- 7 Quantum Annealing, Optimization Algorithms (4 hours)**
- 7.1 Principles of quantum annealing
 - 7.2 Relationship with adiabatic quantum computation
 - 7.3 Optimization problem formulation (QUBO, Ising models)
 - 7.4 Quantum annealer architecture
- 8 Quantum Machine Learning (4 hours)**
- 8.1 Introduction to QML and its importance
 - 8.2 Quantum data representation and feature encoding
 - 8.3 Variational Quantum Circuits (VQC)
 - 8.4 Simple quantum classifiers and QNN concepts
 - 8.5 Hybrid quantum-classical learning models
- Tutorial (30 hours)**
- 1. Classical versus quantum computation; Qubit basics, superposition, and measurement
 - 2. State vectors, normalization and probability amplitudes
 - 3. Dirac notation and Bloch sphere representation with simple simulations
 - 4. Single-qubit gates (X, Y, Z, H, Phase); Matrix representation and simulation
 - 5. Multi-qubit systems and two-qubit gates such as CNOT; Tensor product exercises and simulation
 - 6. Entanglement and Bell states; Measurement correlation analysis
 - 7. Simulation of quantum key distribution protocols, including BB84
 - 8. Quantum algorithms I: Deutsch and Deutsch–Jozsa; Algorithm formulation and coding
 - 9. Quantum algorithms II: Bernstein–Vazirani and Grover search; Coding and simulation exercises
 - 10. Introduction to quantum hardware; Physical qubit types, noise, and decoherence; Basic modeling
 - 11. NISQ devices; Build and execute simple circuits on IBM Quantum or equivalent simulator and analyze results

12. Quantum error modeling including bit-flip and phase-flip; Simple error detection and correction exercises
13. Quantum annealing and optimization; Formulation of QUBO and Ising problems
14. Quantum machine learning and variational circuits; Data encoding, simple VQC construction

Practical

(15 hours)

1. IBM qiskit / quantum composer labs
2. Implement Grover's or teleportation circuit
3. Implement Shor's algorithm
4. Applications in combinatorial optimization (E.g., scheduling, routing, network optimization)
5. Hands-on simulation using quantum annealers
6. Mini-project demonstration

Final Exam

The questions will cover all the chapters in the syllabus. The evaluation scheme will be as indicated in the table below:

Chapter	Hours	Marks distribution*
1	6	8
2	6	8
3	5	7
4	10	12
5	6	8
6	4	6
7	4	5
8	4	6
Total	45	60

* There may be minor deviation in marks distribution.

References

1. Nielsen, M. A., Chuang, I. L. (2002). Quantum computation and quantum information. Cambridge University Press.
2. Bernhardt, C. (2019). Quantum computing for everyone. MIT Press.
3. Schuld, M., Petruccione, F. (2021). Machine learning with quantum computers. Springer.
4. Sahni, V. (2007). Quantum computing. Tata McGraw-Hill Publishing Company.
5. Akama, S. (2014). Elements of quantum computing: History, theories and engineering applications. Springer International Publishing.
6. Wichert, A. (2014). Principles of quantum artificial intelligence. World Scientific Publishing Co.



YEAR III, PART II

Semester 6

6 COURSES IN THIS SEMESTER

ENCE 356	Engineering Economics	130
ENCT 351	Artificial Intelligence	133
ENCT 352	Software Engineering	137
ENCT 353	Simulation and Modeling	141
ENCT 354	Minor Project	145
ENCT 385-399	Elective II	147

6

ENGINEERING ECONOMICS

ENCE 356

Lecture : 3
Tutorial : 1
Practical : 0

Year : III

Part : II

Course Objectives:

The objective of this course is to provide concept of economic principles and the economic environment at the project, firm, societal, and national levels, enabling them to analyze cause-and-effect relationships. The course aims to equip students with the ability to apply economic theories and tools for project selection, equipment replacement, property valuation, and price variation. By the end of the course, students will be able to evaluate alternatives and make informed, economically sound decisions in engineering and business contexts.

1 Introduction (2 hours)

- 1.1 Micro, macro and engineering economics (History, fundamental principle and application)
- 1.2 Terminology related to engineering economics
- 1.3 Economic decision and role of engineers in decision making
- 1.4 Cash flow and cash flow diagram

2 Market Economics (3 hours)

- 2.1 Market, demand, supply and relationship
- 2.2 Elasticity, application of elasticity and government policies
- 2.3 Externality and market inefficiency
- 2.4 Market failure and firm behavior

3 Cost (8 hours)

- 3.1 Cost classification
 - 3.1.1 Total, average, fixed, variable, and marginal costs
 - 3.1.2 Direct, indirect, and standard costs
 - 3.1.3 Cash versus book cost, manufacturing and non-manufacturing cost
 - 3.1.4 Sunk cost, opportunity cost, element of cost, life-cycle cost
- 3.2 Cost estimation and control

4 Time Value of Money (6 hours)

- 4.1 Money (Type, functions and time value of money)
- 4.2 Simple and compound interests (Nominal, effective and continuous compounding)

- 4.3 Economic equivalence
- 4.4 Cash flow types (Single, uniform, linear gradient, geometric gradient and irregular)

5 Methods of Economic Analysis (12 hours)

- 5.1 Capital budgeting
- 5.2 Minimum attractive rate of return (MARR)
- 5.3 Economic analysis of single and multiple projects
 - 5.3.1 Payback period (Simple and discounted)
 - 5.3.2 Equivalent worth (Net present, annual, future and capitalized)
 - 5.3.3 Rate of return (Internal and external)
 - 5.3.4 Public Sector economic analysis (Benefit cost analysis)
 - 5.3.5 Financial and economic analysis
- 5.4 Weighted average cost of capital (WACC)
- 5.5 Repeatability assumption and co-Terminated assumptions
- 5.6 Multiple investment project alternatives (Dependent, independent and contingent)

6 Replacement Analysis (5 hours)

- 6.1 Replacement strategies (Asset life and selection of challengers over defenders)
- 6.2 Economic service life of an asset
- 6.3 Replacement strategy for asset (Project with finite and infinite planning horizon)

7 Risk Analysis (5 hours)

- 7.1 Origin of risk in projects
- 7.2 Risk analysis of projects (Sensitivity, breakeven and scenario analyses)
- 7.3 Decision tree

8 Depreciation and Taxes (5 hours)

- 8.1 Concept and terminology
- 8.2 Depreciation calculation (Straight line, declining balance, sinking fund, sum of the year digit and modified accelerated cost recovery methods)
- 8.3 Tax and corporate income tax
- 8.4 Economic analysis (After-tax cash flow)

9 Measurement of Nation Income (5 hours)

- 9.1 Gross domestic product (Components, real and nominal gross domestic product)
- 9.2 Unemployment (Measurement, job search, minimum wage law)

- 9.3 Inflation
 - 9.3.1 Causes, effects and measurement of inflation
 - 9.3.2 Constant and current cash flow
 - 9.3.3 Equivalence calculation under inflation
 - 9.3.4 Inflation controlling measures
- 9.4 Real and nominal exchange rates, fiscal budget and monetary policy
- 9.5 Financial statement

Tutorial

(15 hours)

1. Cash flow diagram construction, apply and analyze it for different projects
2. Problems related to market equilibrium, elasticity, and government policies
3. Exercises on calculating interests, and analyze cash flows with economic equivalence
4. Application of capital budgeting techniques for given cash flows
5. Selection of best replacement strategies and also, apply risk analysis for real case
6. Computation of after-Tax cash flow for economic analysis
7. Calculation and analysis of gross domestic product, performing equivalence calculations under different inflation scenarios, analyze real and nominal exchange rates
8. Calculation of price variation and price escalation
9. Summarize the fiscal budget

Final Exam

The questions will cover all the chapters in the syllabus. The evaluation scheme will be as indicated in the table below:

Chapter	Hours	Mark distribution*
1	2	3
2	3	5
3	2	3
4	6	8
5	12	16
6	5	6
7	5	6
8	5	8
9	5	5
Total	45	60

* There may be minor deviation in marks distribution.

References

1. Mankiw, N. G. (2017). Principles of economics (8th ed.). Cengage Learning.
2. Park C.S. (2016). Contemporary Engineering Economics: Prentice Hall, Inc.
3. McConnell, C. R., Brue, S. L., Flynn, S. M. (2020). Economics: Principles, problems, and policies (22nd ed.). McGraw-Hill Education.

ARTIFICIAL INTELLIGENCE

ENCT 351

Lecture : 3

Tutorial : 1

Practical : 3

Year : III

Part : II

Course Objectives:

The objective of this course is to provide students with a foundation in Artificial Intelligence (AI), covering intelligent agents, search techniques, knowledge representation, machine learning, and AI ethics. It aims to equip students with both theoretical understanding and practical skills to apply AI techniques to real-world problems, while also developing awareness of the ethical and societal implications of AI systems.

1 Introduction (4 hours)

- 1.1 Definition, foundation, history of AI
- 1.2 Importance of knowledge and learning
- 1.3 Cognition and learning (Neuroscience)
- 1.4 Human intelligence and machine intelligence
- 1.5 AI tree: Branches and interdisciplinary nature
- 1.6 Intelligent agents and types
- 1.7 Agentic AI

2 Problem Solving and Search (9 hours)

- 2.1 Formal problem definition: States, actions, transitions, well-defined problems
- 2.2 Constraint satisfaction problems: Node consistency, path consistency, backtracking
- 2.3 Search algorithms, strategies and evaluations
- 2.4 Uninformed: BFS, DFS, iterative deepening
- 2.5 Informed Search: Best first search, greedy search, A* algorithm
- 2.6 Adversarial search: Minimax algorithm, alpha-beta pruning
- 2.7 Local search and optimization: Hill climbing, simulated annealing
- 2.8 Evolutionary optimization: Genetic algorithm

3 Knowledge Representation and Probabilistic Reasoning (7 hours)

- 3.1 Knowledge-based agent
- 3.2 Knowledge representation techniques and issues in representation
- 3.3 Propositional and predicate logic
- 3.4 Semantic networks, frames and knowledge graph
- 3.5 Review of Bayes' Theorem and probabilistic reasoning

3.6 Fuzzy logic: Membership functions, fuzzy inference systems

4 Machine Learning Fundamentals (10 hours)

- 4.1 Foundations and four pillars of machine learning
- 4.2 Review of mathematics for machine learning
- 4.3 Continuation optimization; Unconstraint optimization, constraint optimization, convex optimization
- 4.4 Review of machine learning pipeline
 - 4.4.1 Model development: Generative versus discriminative
 - 4.4.2 Learning algorithm
 - 4.4.3 Capacity, overfitting and underfitting
 - 4.4.4 Hyperparameters and validation sets
 - 4.4.5 Estimators, bias and variance
 - 4.4.6 Review of MLE and MAP and cross entropy
- 4.5 Supervised learning algorithm: Decision tree and support vector machine
- 4.6 Unsupervised learning algorithm: t-SNE
- 4.7 Semi supervised and reinforcement learning
- 4.8 Model evaluation: Energy-based indicators

5 Neural Networks and Deep Learning Algorithms (8 hours)

- 5.1 Neural networks: Structures, activation functions and universal approximation theorem
- 5.2 Perceptron, multilayer perceptron and backpropagation
- 5.3 Introduction to deep learning
- 5.4 Concepts on recurrent and generative neural networks

6 AI Applications (5 hours)

- 6.1 Expert systems: Characteristics, architecture, development and various applications
- 6.2 NLP: Level of analysis, challenges, modern approaches and applications
- 6.3 Robotics and computer vision: Fundamental, components and applications
- 6.4 Sustainable AI systems

7 Emerging Trends (2 hours)

- 7.1 Sequence to sequence models
- 7.2 Federated learning
- 7.3 Edge AI
- 7.4 Ethics and AI: Responsible AI

Tutorial (15 hours)

- 1. Design a simple agent model (Vacuum cleaner, chess player)
- 2. Formulate a route-finding problem using states, actions and transitions
- 3. Solve the problem with uninformed strategies

4. Apply the A* algorithm with the given heuristic values
5. Solve the minimax tree and demonstrate alpha-beta pruning
6. Solve a CSP like map coloring or mini-Sudoku
7. Perform one iteration of the genetic algorithm (Selection, crossover, mutation)
8. Represent facts using semantic networks and frames
9. Solving problems related to Bayes' theorem and Fuzzy Logic
10. Understanding machine learning basics through model training, cross-validation and performance evaluation
11. Calculation related to decision tree and support vector machine
12. Model Evaluation: Build a confusion matrix; Calculate sensitivity, specificity, precision, recall and f1-score
13. Solve perceptron weight updates for a small dataset
14. Derive forward and backward propagation numerically for a 2-layer NN
15. Case study: Identify ethical risks in real-world AI use-responsible AI in Nepal's context

Practical

(45 hours)

1. Intelligent agents and problem formulation
2. Uninformed search techniques
3. Informed (Heuristic) search techniques
4. Adversarial search and game playing
5. Constraint satisfaction problems (CSP)
6. Evolutionary computation (Genetic algorithms)
7. Knowledge representation (Logic, semantic networks, frames)
8. Probabilistic and fuzzy reasoning
9. Machine learning pipeline and data preprocessing
10. Supervised and unsupervised learning
11. Neural networks and deep learning basics
12. Mini project, AI applications and ethics

Final Exam

The questions will cover all the chapters in the syllabus. The evaluation scheme will be as indicated in the table below:

Chapters	Hours	Marks distribution*
1, 6 and 7	11	14
2	9	12
3	7	10
4	10	14
5	8	10
Total	45	60

* There may be minor deviation in marks distribution.

References

1. Russell, S., Norvig, P. (2021). Artificial intelligence: A modern approach. Pearson.
2. Rich, E., Knight, K., Nair, S. B. (2009). Artificial intelligence. McGraw-Hill Education.
3. Bishop, C. M. (2006). Pattern recognition and machine learning. Springer.
4. Deisenroth, M. P., Faisal, A. A., Ong, C. S. (2020). Mathematics for machine learning. Cambridge University Press.
5. Goodfellow, I., Bengio, Y., Courville, A. (2016). Deep learning. MIT Press.

SOFTWARE ENGINEERING

ENCT 352

Lecture : 3
Tutorial : 1
Practical : 3/2

Year : III
Part : II

Course Objectives:

The objective of this course is to provide students with a foundation in software engineering, covering software characteristics, principles, process models, requirements engineering, system and object-oriented modeling, architectural and design principles, coding standards, testing and quality assurance. Students will gain both theoretical understanding and practical experience in applying software engineering concepts to a project development, including modern practices such as CI/CD, containerization and AI-assisted development.

1 Introduction (4 hours)

- 1.1 Software and software engineering
- 1.2 Nature and characteristics of software
- 1.3 Software application domains
- 1.4 Legacy software
- 1.5 Software crisis
- 1.6 Software myths
- 1.7 Software engineering practice: Essence of practice, general principles

2 The Software Process (8 hours)

- 2.1 Process framework and umbrella activities
- 2.2 Traditional (Plan-driven) process models
 - 2.2.1 Waterfall model and its extensions
 - 2.2.2 Incremental process model
 - 2.2.3 Evolutionary process models (Prototyping, spiral)
- 2.3 Agile and adaptive process models
 - 2.3.1 Agile manifesto and the 12 principles
 - 2.3.2 Agile versus plan-driven development
 - 2.3.3 Scrum framework (Roles, artifacts, ceremonies)
 - 2.3.4 Extreme programming (XP) practices
 - 2.3.5 Lean software development
- 2.4 Model selection considerations

3 Software Requirements Engineering (6 hours)

- 3.1 Requirement engineering process

- 3.2 SRS (Structure, characteristics, users)
- 3.3 Functional and non-functional requirements
- 3.4 Gathering requirements using use cases modeling and scenarios
- 3.5 Agile requirements engineering
 - 3.5.1 User stories and acceptance criteria
 - 3.5.2 Product backlog creation and prioritization
 - 3.5.3 Story mapping basics
 - 3.5.4 Continuous requirements refinement (Backlog grooming)

4 Architectural Design (3 hours)

- 4.1 Introductions and importance
- 4.2 Architectural design principles
- 4.3 Taxonomy of architectural styles
- 4.4 Modular design, cohesion and coupling

5 System Modeling (9 hours)

- 5.1 System modeling
 - 5.1.1 Need for system modeling
 - 5.1.2 Role of abstractions in managing complexity
- 5.2 Process modeling using DFD
- 5.3 Scenario-based analysis
 - 5.3.1 Concept of scenarios
 - 5.3.2 Use-case descriptions and diagrams
- 5.4 Behavioral and structural modeling
 - 5.4.1 Activity diagrams
 - 5.4.2 Class-based modeling

6 Coding and Testing (5 hours)

- 6.1 Coding standards and guidelines
- 6.2 Code review
 - 6.2.1 Code walkthrough
 - 6.2.2 Code inspection
 - 6.2.3 Cleanroom technique
- 6.3 Software testing fundamentals
 - 6.3.1 Verification and validation
 - 6.3.2 Unit testing
 - 6.3.3 Integration testing
 - 6.3.4 System testing
 - 6.3.5 Acceptance testing
- 6.4 Black-box and white-box testing approach
- 6.5 Agile testing practices
 - 6.5.1 Test-driven development (TDD)
 - 6.5.2 Continuous testing

6.5.3 Automated unit testing basics

6.5.4 Refactoring for code quality

7 Software Quality, Assurance, Maintenance (4 hours)

- 7.1 Quality concepts
- 7.2 Quality attributes
- 7.3 Reviews, inspections and QA concepts
- 7.4 ISO standards, CMMI levels
- 7.5 Software maintenance and its types
- 7.6 Maintenance effort and lifecycle considerations

8 Software Configuration Management (3 hours)

- 8.1 Software configuration management
- 8.2 Version control and continuous integration
- 8.3 Change management process
- 8.4 Branching strategies (Git-flow basics)

9 Recent Trends (3 hours)

- 9.1 DevOps and continuous deployment pipelines
- 9.2 Continuous integration/continuous delivery (CI/CD) in modern projects
- 9.3 Cloud-native, microservices architectures and infrastructure as code (IaC)
- 9.4 AI-assisted software development
- 9.5 Low-code/no-code, green software engineering

Tutorial (15 hours)

- 1. Analyze the project domain, identify challenges, and discuss software myths
- 2. Select and justify an appropriate software process model
- 3. Prepare preliminary SRS, including functional and non-functional requirements
- 4. Elicit and model requirements using use cases (Actors, scenarios, system boundary, diagram, and descriptions)
- 5. Develop behavioral models (Scenario identification and activity diagrams)
- 6. Develop a structural model (Class diagram)
- 7. Propose an architectural style and apply modular design principles
- 8. Prepare user stories and product backlog; Conduct a mock sprint planning session
- 9. Define coding standards and perform basic code review/refactoring
- 10. Design black-box and white-box test cases
- 11. Apply version control using git (Commit, branch, merge, conflict resolution)
- 12. Plan a basic CI/CD workflow and compile the final documentation of all artifacts

Practical**(22.5 hours)**

1. Preparation and organization of software requirements specification (SRS) using standard templates
2. Construction of UML diagrams (Use case, activity, class, and sequence) using modeling tools
3. Design of system architecture and component structure using CASE tools
4. Implementation of key system modules following coding standards and best practices
5. Execution of testing procedures (Unit and integration) and documentation of results
6. Application of version control (Git), CI/CD workflow setup, and containerization using docker
7. Term project

Final Exam

The questions will cover all the chapters in the syllabus. The evaluation scheme will be as indicated in the table below:

Chapter	Hours	Marks distribution*
1	4	5
2	8	11
3	6	8
4	3	4
5	9	12
6	5	7
7	4	5
8	3	4
9	3	4
Total	45	60

* There may be minor deviation in marks distribution.

References

1. Pressman, R. S., Maxim, B. R. (2019). Software engineering: A practitioner's approach. McGraw-Hill Education.
2. Mall, R. (2018). Fundamentals of software engineering. PHI Learning.
3. Sommerville, I. (2016). Software engineering. Pearson.
4. Martin, R. C. (2009). Clean code: A handbook of agile software craftsmanship. Pearson.
5. Schwaber, K., Beedle, M. (2002). Agile software development with Scrum. Prentice Hall.

SIMULATION AND MODELING

ENCT 353

Lecture : 3
Tutorial : 1
Practical : 3/2

Year : III
Part : II

Course Objectives:

The objective of this course is to develop knowledge of modeling and simulation techniques for discrete and continuous systems. It emphasizes the development and analysis of simulation models, generation and testing of random numbers and variables, and application of simulation methods to evaluate the performance of stochastic systems.

1 Introduction to Simulation (4 hours)

- 1.1 System and system environment concepts
- 1.2 Continuous and discrete systems
- 1.3 Types of models
- 1.4 Model development life cycle
- 1.5 Simulation and steps in simulation study
- 1.6 Advantages and disadvantages of simulation
- 1.7 Monte-Carlo simulation
- 1.8 Discrete-event system simulation

2 Physical and Mathematical Models (4 hours)

- 2.1 Differential and partial differential equations
- 2.2 Static physical model
- 2.3 Dynamic physical model
- 2.4 Static mathematical models
- 2.5 Dynamic mathematical models

3 Simulation of Continuous System (5 hours)

- 3.1 Continuous system models
- 3.2 Analog computer
- 3.3 Analog methods
- 3.4 Hybrid simulation
- 3.5 Digital-analog simulators
- 3.6 Continuous system simulation languages (CSSLs)
- 3.7 Feedback systems

4 Simulation of Queuing System (6 hours)

- 4.1 Elements of queuing system

- 4.2 Characteristics of queuing systems
- 4.3 Model of queuing system
- 4.4 Types of queuing system
- 4.5 Queuing notation (Kendall's notation)
- 4.6 Measurement of system performance
- 4.7 Network of queues
- 4.8 Applications of queuing system

5 Markov Chains (3 hours)

- 5.1 Key features of Markov chains
- 5.2 Markov process with examples
- 5.3 Applications of Markov chains

6 Random Number (10 hours)

- 6.1 Properties of random numbers
- 6.2 Generation of pseudo-random numbers
- 6.3 Random number generation: Linear and arithmetic congruential methods
- 6.4 Tests for random numbers
 - 6.4.1 Kolmogorov-Smirnov test
 - 6.4.2 Chi-Square (χ^2) test
 - 6.4.3 Gap test
 - 6.4.4 Poker's method
 - 6.4.5 Testing for auto correlation
- 6.5 Generating discrete distribution
- 6.6 Inversion, rejection, composition and convolution

7 Verification and Validation of Simulation Models (3 hours)

- 7.1 Verification and validation
- 7.2 Verification of simulation models
- 7.3 Calibration and validation of models
- 7.4 Naylor and finger validation process
- 7.5 Validation: Errors

8 Analysis of simulation output (4 hours)

- 8.1 Confidence intervals and hypothesis testing
- 8.2 Estimation methods
- 8.3 Simulation run statistics
- 8.4 Replication of runs
- 8.5 Elimination of initial bias

9 Simulation software (3 hours)

- 9.1 Simulation in Java

- 9.2 Simulation in GPSS
- 9.3 Simulation in Python
- 9.4 Other simulation software

10 Simulation of Computer Systems (3 hours)

- 10.1 Simulation tools
- 10.2 High level computer: System simulation
- 10.3 CPU simulation
- 10.4 Memory simulation
- 10.5 Simulation of computer networks

Tutorial (15 hours)

1. Poisson process
2. Monte Carlo simulation
3. Queuing system and Markov chains
4. Different methods of random number generation
5. Kolmogorov-Smirnov test, Chi-square test, gap-test, poker and autocorrelation test
6. Modeling continuous and discrete systems
7. Simulation output analysis

Practical (22.5 hours)

1. Simulation of the R-C amplifier circuit and mass spring damper system
2. Generation of random number
3. Chi-square goodness-of-fit test and Kolmogorov-Smirnov test
4. Simulation of queuing system
5. Simulation of Markov chain

Final Exam

The questions will cover all the chapters in the syllabus. The evaluation scheme will be as indicated in the table below:

Chapters	Hours	Marks distribution*
1, 2	8	10
3	5	7
4	6	8
5	3	5
6	10	13
7	3	4
8	4	5
9, 10	6	8
Total	45	60

* There may be minor deviation in marks distribution.

References

1. Banks, J. (2005). Discrete-event system simulation. Pearson.
2. Gordon, G. (1978). System simulation (Latest Edition). Prentice Hall.
3. Law, A. M., Kelton, W. D. (2007). Simulation modeling and analysis. McGraw-Hill Education.
4. Rubinstein, R. Y., Melamed, B. (1998). Modern simulation and modeling (Latest Edition). John Wiley & Sons.
5. Singh, V. P. (2009). System modeling and simulation. New Age International Publishers.

MINOR PROJECT

ENCT 354

Credit :1

Year: III

Part: II

Course Objectives:

The objective of this course is to enable students to carry out a small-scale engineering research- or product-based project in the field of computer engineering, thereby developing practical skills and hands-on experience in system design, implementation, and testing

General Procedures

Under the supervision of an assigned supervisor, students shall be organized into groups of 3–4 members. Each group shall select a project topic related to computer engineering, with emphasis on areas such as artificial intelligence and machine learning, image processing and computer vision, networks and cybersecurity, audio and natural language processing, electronic devices and embedded systems, big data technologies, Internet of Things (IoT), and web and mobile application development. Where appropriate, students may conduct field visits or collect real-world datasets for system design, implementation, and analysis. Each group shall systematically complete the project through stages including topic selection, planning, development, testing, documentation, and final presentation. Continuous guidance, monitoring, and feedback shall be provided by the supervisor throughout the duration of the course to ensure effective project development and attainment of learning outcomes.

1 Project Selection and Planning

- 1.1 Selection of minor project based on theory/practical subject studied or as per the research or industry needs
- 1.2 Problem identification and scope definition
- 1.3 Approval of project by the department

2 Literature Review and Proposal

- 2.1 Basic literature review and background study, study of existing systems in case of product based project
- 2.2 Preparation of brief project proposal including objective and methodology
- 2.3 Selection of software tools, technologies, and resources (Programming languages, frameworks, databases)

3 Design, Development and Project Management

- 3.1 Design and development (Software/hardware/hybrid system)

- 3.2 Application of programming skills, algorithms and engineering tools
- 3.3 Integration of concepts from different subjects (Computer programming, object oriented programming, software engineering, artificial intelligence, compiler design, big data technology, database management system, computer organization and architecture, computer network, electronic devices)
- 3.4 Applications of project management skills such as work division, time management, leadership

4 Implementation and Testing

- 4.1 Implementation and deployment of the project
- 4.2 Testing and validation of the developed system (Unit testing, integration testing, and user testing)
- 4.3 Performance evaluation and improvement

Timeline of the defense

- i. Proposal submission and defense: A group of students shall submit the proposal in the prescribed format to the concerned department within two weeks after the commencement of the sixth semester
- ii. Mid-term defense: The mid-term defense of the minor project shall be conducted one month prior to the final defense
- iii. Final defense: The final defense shall be conducted two weeks prior to the board examination

Evaluation

The final evaluation of the minor project shall be based on assessments from the supervisor, the proposal defense, mid-term defense, final defense, and the internal examiner. The respective weightage and total marks are presented in the following table:

Supervisor	Defense			Internal Examiner	Total Marks
	Proposal	Mid-Term	Final		
25	5	5	5	10	50

NETWORK AND SYSTEMS PROGRAMMING

ENCT 386

Lecture : 3

Tutorial : 2

Practical : 1

Year : III

Part : II

Course Objectives:

The objective of this course is to provide knowledge of networking and system-level programming concepts. It focuses on socket programming, process and thread management, and the use of system calls for efficient resource handling. Upon completion, students will be able to develop concurrent, scalable, and high-performance network applications suitable for real-world computing environments.

1 System Programming Fundamentals (4 hours)

- 1.1 Overview of system programming and its role in networking
- 1.2 User space versus kernel space
- 1.3 System calls and standard libraries
- 1.4 File descriptors and low-level I/O
- 1.5 Linux/Unix system architecture overview

2 Process Management and IPC (7 hours)

- 2.1 Process creation and control (Fork, exec, wait)
- 2.2 Process states and scheduling concepts
- 2.3 Signals and signal handling
- 2.4 Inter-process communication: Pipes and FIFOs, message queues, shared memory, semaphores

3 Threads and Concurrent Programming (7 hours)

- 3.1 Thread models and lifecycle
- 3.2 POSIX threads (Pthreads)
- 3.3 Thread synchronization: Mutexes, condition variables
- 3.4 Deadlocks, race conditions, and avoidance
- 3.5 Multithreading versus multiprocessing (Performance considerations)

4 Socket Programming and Network APIs (10 hours)

- 4.1 Socket interface and API
- 4.2 TCP socket programming (Client-server model)
- 4.3 UDP socket programming
- 4.4 Blocking versus non-blocking sockets
- 4.5 I/O multiplexing (Select, poll basics)

4.6 Error handling and debugging techniques

5 Advanced Network Programming (7 hours)

- 5.1 Concurrent server design: Iterative versus concurrent servers, multi-process and multi-threaded servers
- 5.2 Event-driven programming: Select, poll, epoll
- 5.3 High-performance server design principles
- 5.4 Concepts of remote procedure call (RPC)
- 5.5 Introduction to RESTful APIs and microservices
- 5.6 Secure communication basics (SSL/TLS overview)

6 System-Level I/O and Performance Optimization (5 hours)

- 6.1 Advanced file I/O (Read/write optimization)
- 6.2 Non-blocking I/O and asynchronous I/O concepts
- 6.3 Memory-mapped files (Mmap)
- 6.4 File locking and concurrency
- 6.5 Performance tuning and profiling basics

7 Network Application Development and Case Studies (5 hours)

- 7.1 Design of scalable client-server systems
- 7.2 Development of network applications: Chat servers, file transfer systems
- 7.3 Debugging and monitoring tools (Netstat, ss, tcpdump, Wireshark)
- 7.4 Case studies of real-world systems (Web servers, distributed apps)

Tutorial (30 hours)

- 1. Implement file operations using low-level system calls (Open, read, write, close) and compare with standard I/O
- 2. Demonstrate file descriptor manipulation using dup()/dup2() for I/O redirection
- 3. Write a program using fork() and exec() to create and replace processes
- 4. Implement IPC using pipes and shared memory, comparing their behavior
- 5. Create multiple threads using POSIX threads (pthreads) and observe execution behavior
- 6. Solve the producer-consumer problem using mutexes and condition variables
- 7. Develop a TCP client-server application (Echo service)
- 8. Implement a UDP communication program and compare with TCP behavior.
- 9. Build a multi-client concurrent server using threads or processes
- 10. Implement I/O multiplexing using select() or poll() for handling multiple clients
- 11. Demonstrate memory-mapped file (mmap) usage and compare with standard file I/O
- 12. Implement file locking mechanisms to handle concurrent file access
- 13. Develop a multi-client chat application using sockets
- 14. Capture and analyze traffic of your application using tools like Wireshark/tcpdump

15. Design and implement a scalable network application (chat/file server) incorporating: Concurrency (threads/processes), socket programming, basic performance optimization

Practical

(15 hours)

1. System calls and file operations in Linux
2. Process creation using fork/exec and process control
3. IPC using pipes, shared memory, and message queues
4. Signal handling programs
5. Thread creation and synchronization using Pthreads
6. TCP client-server socket programming
7. UDP-based communication programs
8. Multi-client concurrent server (thread/process-based)
9. I/O multiplexing using select/poll
10. Packet capture and analysis using Wireshark
11. Mini-project: Design and implement a scalable network application

Final Exam

The questions will cover all the chapters in the syllabus. The evaluation scheme will be as indicated in the table below:

Chapter	Hours	Marks distribution*
1	4	5
2	7	9
3	7	9
4	10	14
5	7	9
6	5	7
7	5	7
Total	45	60

* There may be minor deviation in marks distribution.

References

1. Forouzan, B. A. (2017). Data communications and networking. McGraw-Hill Education.
2. Kurose, J. F., Ross, K. W. (2021). Computer networking: A top-down approach. Pearson.
3. Stevens, W. R. (2013). Advanced programming in the UNIX environment. Addison-Wesley.
4. Stevens, W. R. (2003). UNIX network programming. Addison-Wesley.
5. Kernighan, B. W., Pike, R. (1984). The UNIX programming environment (Latest Edition). Prentice Hall.

Next Generation Networking with IPv6

ENCT 387

Lecture : 3

Tutorial : 2

Practical : 1

Year : III

Part : II

Course Objectives:

The objective of this course is to develop a solid understanding of internet protocols with a focus on IPv6 and next-generation networking concepts. It aims to equip students with knowledge of IPv6 addressing, routing, security and transition mechanisms from IPv4 to IPv6. The course also emphasizes practical skills in configuring, analyzing and deploying IPv6-based networks and services. Ultimately, it prepares students to design and manage modern network infrastructures in real-world environments.

1 Fundamentals of Internet and Networking Protocols (3 hours)

- 1.1 Evolution of the Internet and networking paradigms
- 1.2 OSI versus TCP/IP model
- 1.3 IPv4 addressing overview
- 1.4 IPv4 addressing review and limitations
- 1.5 Need and importance of next generating IP addressing

2 Next Generation Internet Protocol (12 hours)

- 2.1 Introduction to IPv6
 - 2.1.1 History and need for IPv6
 - 2.1.2 IPv6 header format
 - 2.1.3 Features of IPv6
 - 2.1.4 IPv6 addressing format and types
 - 2.1.5 Extension headers
- 2.2 ICMPv6
 - 2.2.1 Features
 - 2.2.2 General message format
 - 2.2.3 ICMP error and informational message types
 - 2.2.4 Neighbor discovery
 - 2.2.5 Path MTU discovery

3 Security and Quality of Service in IPv6 (6 hours)

- 3.1 Network threats and vulnerabilities in IPv6
- 3.2 Security techniques
- 3.3 IPSEC framework
- 3.4 Basic firewall and filtering concepts

- 3.5 Zero trust networking
- 3.6 QoS concept in IPv6
- 3.7 QoS in multimedia and real-time applications

4 Routing with IPv6 (6 hours)

- 4.1 Routing in the internet and CIDR
- 4.2 Unicast and multicast routing concepts
- 4.3 Unidirectional link routing
- 4.4 Routing protocols: RIPng, OSPFv3 and BGP (Basic concepts for IPv6)
- 4.5 Multicast routing (PIM-SM overview)

5 IPv4/IPv6 Transition Mechanisms (6 hours)

- 5.1 Tunneling: Automatic tunneling; Configured tunneling
- 5.2 Dual stack approach
- 5.3 Translation mechanisms
- 5.4 Migration strategies for enterprises and ISPs
- 5.5 Case studies of global IPv6 transition

6 IPv6 Deployment and Network Design (5 hours)

- 6.1 IPv6 deployment challenges and risks
- 6.2 Enterprise and ISP deployment strategies
- 6.3 IPv6 addressing plan design
- 6.4 IPv6 DNS (AAAA records)
- 6.5 IPv6 enabled proxy, web and mail servers

7 Advanced Networking Applications (7 hours)

- 7.1 Fundamentals of SDN
- 7.2 Introduction to IBN and NDN
- 7.3 IPv6 as the foundations of 5G network
- 7.4 IPv6 integration SDN/IBN/NDN

Tutorial (30 hours)

- 1. Compare OSI vs TCP/IP models through layered packet flow analysis
- 2. Analyze limitations of IPv4 addressing and justify the need for IPv6 using numerical examples
- 3. Construct and analyze an IPv6 packet, including header fields and extension headers
- 4. Perform IPv6 addressing and subnetting exercises (Global, link-local, multicast)
- 5. Implement and observe neighbor discovery protocol and ICMPv6 message types
- 6. Demonstrate basic IPsec configuration concepts
- 7. Analyze QoS mechanisms in IPv6, especially for real-time/multimedia traffic

8. Configure and test IPv6 unicast routing (OSPFv3 basics) in a simulated environment
9. Comparison of IPv6 multicast routing (PIM-SM concept) with unicast routing
10. Implement dual stack configuration in a lab setup
11. Demonstrate tunneling mechanisms and analyze packet flow
12. Design an IPv6 addressing plan for an enterprise network
13. Configure and test IPv6 DNS (AAAA records) and basic web service
14. Analyze the role of IPv6 in SDN/5G/NDN architectures through a case study
15. Design and simulate a complete IPv6-enabled network, including: Addressing plan, routing (OSPFv3 or static), transition mechanism (Dual stack/tunneling), basic service deployment (DNS/Web)

Practical

(15 hours)

1. Enabling and configuring IPv6 in Windows/Linux
2. IPv6 header and packet analysis using Wireshark
3. Neighbor Discovery and ICMPv6 analysis
4. IPv6 addressing and subnetting practice
5. Unicast routing using OSPFv3
6. Multicast routing using PIM-SM
7. IPv6 DNS, web, and proxy server configuration
8. Case Study: Design and deploy an IPv6-based network

Final Exam

The questions will cover all the chapters in the syllabus. The evaluation scheme will be as indicated in the table below:

Chapters	Hours	Marks distribution*
2	12	16
3	6	8
4	6	8
5	6	8
6	5	8
1 and 7	10	12
Total	45	60

* There may be minor deviation in marks distribution.

References

1. Davies, J. (2008). Understanding IPv6. Microsoft Press.
2. Hagen, S. (2014). IPv6 essentials. O'Reilly Media.
3. Thomas, S. A. (1996). IPng and the TCP/IP protocols. John Wiley & Sons.
4. Hersent, O., Gurle, D., & Petit, J.-P. (2000). IP telephony: Deploying VoIP protocols. Addison-Wesley.

ANALYSIS OF ALGORITHMS

ENCT 388

Lecture : 3

Tutorial : 2

Practical : 1

Year : III

Part : II

Course Objectives:

The objective of this course is to provide students with a strong foundation in the analysis of algorithm efficiency and computational complexity. It aims to develop understanding of key algorithm design paradigms, including divide-and-conquer, greedy methods, dynamic programming, and backtracking, and covers fundamental concepts of NP-completeness and approximation algorithms.

1 Introduction to Algorithm Analysis (5 hours)

- 1.1 Algorithm and its properties, RAM model, time and space complexity, detailed analysis of algorithms, Concept of Aggregate Analysis
- 1.2 Asymptotic notations (Big-O, Big- Ω and Big- Θ), their geometrical interpretations and examples
- 1.3 Concept of best case, average case and worst case performance of an algorithm
- 1.4 Modeling algorithms by recurrence relation
- 1.5 Solving recurrence relation for evaluating computational complexity
 - 1.5.1 Recursion tree method
 - 1.5.2 Substitution method
 - 1.5.3 Using masters theorem

2 Iterative and Numeric Algorithms (8 hours)

- 2.1 Algorithm for GCD and Fibonacci number
- 2.2 Sequential search
- 2.3 Review of bubble sort, selection sort, and insertion sort algorithms
- 2.4 Number theoretic notations
- 2.5 Euclid's and Extended Euclid's algorithms
- 2.6 Solving modular linear equations using Chinese remainder theorem
- 2.7 Fermat's theorem
- 2.8 Miller-Rabin randomized primality test and algorithm

3 Divide and Conquer Algorithms (8 hours)

- 3.1 Binary search, min max finding algorithm
- 3.2 Analysis of sorting algorithms
 - 3.2.1 Merge sort

- 3.2.2 Heap sort
- 3.2.3 Quick sort
- 3.2.4 Randomized quick sort
- 3.3 Order statistics
 - 3.3.1 Selection in expected linear time
 - 3.3.2 Selection in worst case linear time

4 Greedy Algorithms (7 hours)

- 4.1 Basic concepts
- 4.2 Fractional knapsack problem
- 4.3 Job sequencing with deadlines
- 4.4 Analysis of minimum spanning trees related algorithms
- 4.5 Analysis of single source shortest path algorithm

5 Dynamic Programming (8 hours)

- 5.1 Basic concepts
- 5.2 All pair shortest path algorithm
- 5.3 Travelling salesperson problem
- 5.4 String editing
- 5.5 0/1 knapsack problem using dynamic programming
- 5.6 Matrix chain multiplication
- 5.7 Flow shop scheduling

6 Backtracking Techniques (4 hours)

- 6.1 Basic concepts
- 6.2 The N-Queen problem
- 6.3 Sum of subsets
- 6.4 Graph coloring
- 6.5 Hamiltonian cycles
- 6.6 0/1 knapsack problem using backtracking approach

7 NP-Hard and NP-Complete Problems (5 hours)

- 7.1 Basic concepts
- 7.2 Cook's theorem
- 7.3 NP-Hard graph problems
- 7.4 NP-Hard scheduling problems
- 7.5 NP-Hard code generation problems
- 7.6 Simplified NP-hard problems
- 7.7 Approximation algorithms: ϵ - approximation, polynomial time approximation scheme, probabilistically good algorithms
- 7.8 Vertex cover problem, subset sum problem

Tutorial**(30 hours)**

1. Algorithm analysis
2. Iterative and numeric algorithms
3. Divide and conquer algorithms
4. Greedy algorithms
5. Dynamic programming
6. Backtracking techniques
7. NP-Hard and NP-complete problems

Practical**(15 hours)**

1. Implementation and complexity analysis of iterative, numeric and recursive algorithms
2. Implementation and complexity analysis of greedy algorithms
3. Implementation and complexity analysis of algorithms involving divide and conquer strategy
4. Implementation and complexity analysis of algorithms based on dynamic programming
5. Implementation and complexity analysis of algorithms using backtracking concept

Final Exam

The questions will cover all the chapters in the syllabus. The evaluation scheme will be as indicated in the table below:

Chapter	Hours	Marks distribution*
1	5	6
2	8	10
3	8	11
4	7	9
5	8	12
6	4	5
7	5	7
Total	45	60

* There may be minor deviation in marks distribution.

References

1. Horowitz, E., Sahni, S., Rajasekaran, S. (2007). Fundamentals of computer algorithms. Universities Press.
2. Cormen, T. H., Leiserson, C. E., Rivest, R. L., Stein, C. (2022). Introduction to algorithms. MIT Press.
3. Kleinberg, J., Tardos, É. (2006). Algorithm design. Pearson.
4. Skiena, S. S. (2020). The algorithm design manual. Springer.
5. Dasgupta, S., Papadimitriou, C. H., Vazirani, U. V. (2006). Algorithms. McGraw-Hill Education.

AUDIO PROCESSING

ENCT 389

Lecture : 3

Tutorial : 2

Practical : 1

Year : III

Part : II

Course Objectives:

The objective of this course is to develop competency in analyzing and processing audio signals in time and frequency domains. It covers fundamentals of acoustics, digital audio representation, spectral analysis and audio effects, along with practical skills in feature extraction and Music Information Retrieval (MIR). Hands-on use of Digital Audio Workstations (DAWs), MIDI, and plugins enables application of audio signal processing techniques in multimedia, speech processing, music technology and emerging digital audio applications.

- 1 Fundamental of Sound and Audio Signal (7 hours)**
 - 1.1 Physical acoustic: Wave propagation, frequency, amplitude, phase
 - 1.2 Psycho acoustic: Pitch, loudness, timbre, dynamics and intensity
 - 1.3 Fundamental of digital audio
 - 1.4 Audio file format: WAV, AIFF, FLAC, ALAC, MP3
 - 1.5 Practical audio quality: Sample rates (44.1khz, 48khz, 96khz) and bit depths
- 2 Time Domain Audio Processing (8 hours)**
 - 2.1 Discrete time audio signal, basic signal operations, system in time domain
 - 2.2 Convolution and impulse response
 - 2.3 Analog and digital filters in audio systems
 - 2.4 Temporal feature extraction
- 3 Spectral Representation (7 hours)**
 - 3.1 DFT and FFT: Derivation and computational complexity
 - 3.2 Windowing function in audio
 - 3.3 Short time Fourier transform: Frame size, hop size and overlap
 - 3.4 Spectrograms: Reading and interpreting time-frequency representation,
 - 3.5 Mel spectrogram: Reading and interpreting; Mel frequency representation
 - 3.6 Constant Q transform and its relevance to music analysis
- 4 Frequency Domain and Non-Linear Effects (10 hours)**
 - 4.1 Sinusoidal model
 - 4.2 Deterministic plus residual model
 - 4.3 Deterministic plus stochastic model

- 4.4 Basis of non-linear effects
- 4.5 Dynamic range control: Envelop follow, compressor and limiting, expansion and gating, ADSR, modulation effect
- 4.6 Time segment process: Time scratching and pitch shifting

5 Music Information Retrieval (6 hours)

- 5.1 MIR overview: Types of music data, challenges, application
- 5.2 Spectral centroid and roll off
 - 5.2.1 Basics of spectral centroid and spectral roll
 - 5.2.2 Comparison between centroid and roll off
 - 5.2.3 Application (Timbre analysis, music/ speech classification)
- 5.3 Mel frequency cepstral coefficient (MFCC)
 - 5.3.1 Motivation for cepstral features
 - 5.3.2 Steps in MFCC
 - 5.3.3 Interpretation of coefficient
 - 5.3.4 Application (Speech recognition, music classification)
- 5.4 Chroma
 - 5.4.1 Concept of pitch classes (12- tone system) mapping
 - 5.4.2 Chromagram representation,
- 5.5 Onset and tempo and their applications

6 Audio system and DAW Integration (4 hours)

- 6.1 Digital audio workstation (DAW)
 - 6.1.1 Role in audio production
 - 6.1.2 History and evolution of DAW
- 6.2 DAW available in market, DAW interface, timeline and arrangement view
- 6.3 MIDI: Basics, MIDI message structure, VSTi basics,
- 6.4 Common plugin: Equalizer, compressor, reverb, delay

7 Recent Trend (3 hours)

- 7.1 Machine learning in audio
- 7.2 Audio foundation models for bioacoustic and healthcare
- 7.3 Spatial audio
- 7.4 Audio DSP ICs
- 7.5 Generative AI in music

Tutorial (30 hours)

1. Discuss lossless versus lossy compression in the context of audio
2. Derive the convolution sum for a linear time-invariant (LTI) system and explain its physical significance in audio signal processing.
3. Discuss different types of analog filter. Write an advantage of FIR and IIR filter in audio processing
4. Case study discussion
5. Explain the various steps involved in MFCC

6. State the Nyquist theorem and find the Nyquist rate for a signal with maximum frequency 8 kHz.
7. Explain the concept of a chromagram in music information retrieval
8. Explain binaural rendering using head-related transfer functions (HRTFs) and identify one application domain for dedicated audio DSP hardware
9. Music feature analyzer: Input (Audio file), output (Spectrogram, MFCC, tempo)
10. Mini DAW project: Record and edit audio using DAW, apply multiple effects, mix a short track
11. Music classifier: Genre basis classifier

Practical

(15 hours)

1. Generate sinusoids; Downsample to observe aliasing
2. Measure file size, SNR, and spectral differences between WAV, MP3 (128k/320k), and FLAC
3. Temporal features extraction
4. Spectrogram and STFT
5. Mel spectrogram and CQT
6. MIR features: MFCC
7. DAW: Implement basic features
8. DAW integration and MIDI, VSTi implementation

Final Exam

The questions will cover all the chapters in the syllabus. The evaluation scheme will be as indicated in the table below:

Chapter	Hours	Marks distribution*
1	7	9
2	8	11
3	7	9
4	10	13
5	6	8
6	4	6
7	3	4
Total	45	60

* There may be minor deviation in marks distribution.

References

1. Christensen, M. G. (2019). Introduction to audio processing. Springer.
2. Smith, J. O. (2011). Spectral audio signal processing. W3K Publishing.
Zölzer, U. (Ed.). (2011). DAFX: Digital audio effects. Wiley.
3. Müller, M. (2015). Fundamentals of music processing: Audio, analysis, algorithms, applications. Springer.
4. Lyons, R. G. (2011). Understanding digital signal processing. Pearson Education.



YEAR IV

Final year

IOE has named these courses in the curriculum structure but has not yet published their syllabi. They will be added to this handbook once released.

Semester 7 YEAR IV, PART I

ENEX 416	Digital Signal Analysis and Processing	Awaiting IOE
ENCT 411	Distributed and Cloud Computing	Awaiting IOE
ENCT 412	ICT Project Management	Awaiting IOE
ENEX 417	Energy, Environment and Social Engineering	Awaiting IOE
ENCT 435-444	Elective III	Awaiting IOE
ENCT 413	Project I	Awaiting IOE

Semester 8 YEAR IV, PART II

ENCT 463	Network and Cyber Security	Awaiting IOE
ENCT 465-474	Elective IV	Awaiting IOE
ENCT 462	Internship	Awaiting IOE
ENCT 461	Project II	Awaiting IOE

IV



Imperial College of Engineering

Affiliated to Tribhuvan University · Bhaktapur, Nepal

icenepal.com

01-5091616

Admissions: icenepal.com/apply