SHIV NADAR UNIVERSITY
CHENNAI

School of Engineering

Department of Computer Science and Engineering

B. Tech
Artificial Intelligence and Data Science

SYLLABUS

SEMESTER 1

COMMUNICATIVE ENGLISH

L	T	P	C
2	1	0	3

COURSE OBJECTIVES:

- To help learners develop the basic **reading** skills as required for academic purposes
- To help learners develop the **writing** abilities as required in academic contexts
- To help learners develop their **listening** skills, which will enable them to listen to lectures and comprehend them by asking questions and seeking clarifications
- To help learners develop their **speaking** skills and speak fluently in real contexts
- To help learners develop **vocabulary**, as required in academic contexts
- To help learners gain the expertise required in **grammar** for them to function well in academic contexts

SHARING INFORMATION RELATED TO ONESELF/FAMILY & FRIENDS: **12**

Reading: Short comprehension passages, Practice in skimming, scanning and predicting
Writing: Completing sentences, Developing hints
Speaking: Introducing oneself, Exchanging personal information
Listening: Listening comprehension of short texts, including formal and informal conversations
Language Development: Asking and answering - Wh- Questions and Yes/No questions
Vocabulary Development: Prefixes and Suffixes, Polite Expressions

GENERAL READING AND FREE WRITING: **12**

Reading: Reading short narratives and descriptions from newspapers including dialogues and conversations
Writing: Paragraph writing (topic sentence, main ideas, organization, cohesive devices)
Listening: Telephonic conversations, short presentations and TV news
Speaking: Sharing information of a personal kind, Greeting, Taking leave
Language development: Prepositions, Conjunctions, Clauses
Vocabulary development: Guessing meanings of words in context

GRAMMAR AND LANGUAGE DEVELOPMENT: **12**

Reading: Short texts and longer passages (close reading)
Writing: Understanding text structure (Use of reference words and discourse markers, coherence markers, reordering jumbled sentences)
Listening: Listening to TED talks and longer texts, product description, and narratives from different sources
Speaking: Asking about routine actions and expressing opinions, Making short presentations
Language development: Degrees of comparison, Pronouns, Direct vs indirect speech
Vocabulary development: Idioms and phrases, Single word substitutes, Adverbs.

READING AND LANGUAGE DEVELOPMENT: **9**

Reading: Reading longer texts and different types of texts (journalistic, literary)
Writing: Letter writing (informal or personal letters), E-mails (conventions of personal email)
Listening: Listening to dialogues or conversations and completing exercises based on them
Speaking: Speaking about oneself, Speaking about one's friend, Role-plays
Language development: Tenses (simple and continuous)
Vocabulary development: Synonyms, Antonyms, Phrasal verbs

EXTENDED WRITING:

Reading: Longer academic texts including comparison and contrast ones
Writing: Brainstorming, developing an outline, and identifying main and subordinate ideas, Dialogue writing, Writing short essays
Listening: Listening to talks and lectures
Speaking: Participating in conversations and short group discussions
Language development: Modal verbs, Perfect and perfect continuous tenses
Vocabulary development: Collocations, Fixed and semi-fixed expressions

TOTAL PERIODS: 45

COURSE OUTCOMES:

On successful completion of this course, the learners will be able to

- Apply reading strategies to comprehend articles of a general kind (ex. magazines and newspapers)
- Participate effectively in informal conversations; introduce themselves and their friends and express opinions in English
- Comprehend conversations and short talks (formal and informal)
- Write short general essays and personal letters and emails

TEXT BOOK:

1. Board of Editors. *Using English: A Course Book for Undergraduate Engineers and Technologists*. Orient BlackSwan Limited, Hyderabad, 2015.

LINEAR ALGEBRA

L	T	P	C
3	1	0	4

COURSE OBJECTIVES:

The objectives of this course are to enable the students to

- Find the basis and dimension of vector space
- Obtain the matrix of linear transformation
- Find the eigenvalues and eigenvectors of the transformations
- Find orthonormal basis of inner product space
- Perform matrix decomposition and to find least square approximation

VECTOR SPACES: **12**

Semigroup - Group - Ring - Field (Definitions and examples) - Vector Space: Subspace - Linear Independence and Dependence - basis and dimension

LINEAR TRANSFORMATION: **12**

Linear Transformation - Range Space and Null Space - Rank and nullity - Dimension Theorem

EIGEN VALUES AND EIGEN VECTORS: **12**

Matrix representation of linear transformation - Eigenvalues and Eigenvectors of Linear Transformation

INNER PRODUCT SPACES: **12**

Inner product and Norms-properties - Orthogonal, Orthonormal Vectors - Gram Schmidt Orthonormalization process

MATRIX DECOMPOSITION: **12**

QR decomposition - Singular Value Decomposition -Least square approximations

TOTAL PERIODS: 60

COURSE OUTCOMES:

After the completion of the course the student will be able to

- Find the basis and dimension of vector space
- Obtain the matrix of linear transformation
- Find the eigenvalues and eigenvectors of linear transformations
- Find orthonormal basis of inner product space
- Apply matrix decomposition in engineering and find least square approximations to the system of equations

TEXT BOOK:

1. Friedberg A.H, Insel A.J. and Spence L, Linear Algebra, Prentice Hall of India, New Delhi, 2004.

ENGINEERING PHYSICS

L	T	P	C
3	0	0	3

COURSE OBJECTIVES:

Enable the students to

- Understand the characteristics of sound; production and applications of ultrasound.
- Develop an understanding of quantum mechanical concepts and their theories.
- Explain the theories of physics of semiconductors.
- Describe the principle of laser action and their production.
- Analyse the propagation of light through optical fibres and losses in fibre optic communication.

ACOUSTICS: **4**

Classification: Music & Noise - Characteristics of Sound Pitch/Frequency, Loudness/Intensity - decibel scale - Weber-Fechner law - Loudness Curves - Quality/Timbre

ULTRASONICS: **5**

Production: Magnetostriction and Piezoelectric methods - Detection: Piezoelectric, Acoustic grating - Non-Destructive Testing - Pulse echo system - Reflection and transmission modes - Modes of data presentation - A, B and C scan displays - Sonogram.

QUANTUM PHYSICS: **9**

Planck's theory (derivation) - Deduction of Wien's displacement law and Rayleigh-Jeans law from Planck's theory - Properties of Matter waves - wave particle duality - Schrödinger's wave equation - Time-independent and time-dependent equations - Physical significance of wave function - Particle in a one dimensional box and extension to three dimensional box - Degeneracy of electron energy states - Quantum free electron theory - Density of states - Fermi-Dirac statistics - Free electron concentration in metals.

SEMICONDUCTORS: **9**

Classification of semiconductors based on doping and band gap - Intrinsic semiconductor - Concept of hole - Carrier concentration derivation - Fermi level and its variation with temperature - Electrical conductivity - Band gap determination - Extrinsic semiconductors - Carrier concentration derivation in n-type and p-type semiconductors - Variation of Fermi level with temperature and impurity concentration.

LASERS: **9**

Interaction of Radiation with Matter - Spontaneous and stimulated emissions - Einstein's A and B coefficients - Conditions for Laser action - Population inversion - Active medium - pumping schemes - Optical resonant cavity - Light Amplification -Types of lasers - Nd: YAG, CO_2 and Semiconductor lasers - Homo junction & hetero junction laser.

FIBRE OPTICS: **9**

Principle and propagation of light in optical fibres - Numerical aperture and Acceptance angle, Types of optical fibres (material, mode & refractive index) - Losses in fibres - Attenuation, dispersion - Fibre Optical Communication system (Block diagram) - Active and passive sensors - pressure, strain, displacement.

TOTAL PERIODS: 45

COURSE OUTCOMES:

At the end of this course, the students will be able to

- Describe the characteristics of sound and Ultrasonics production and applications
- Explain the basic quantum mechanical concepts and their applications
- Analyse the physics of semiconductors
- Elucidate the principle and working of different type of lasers
- Explicate the principle, propagation and losses in fibre optic communication

TEXT BOOK:

1. M. N. Avadhanulu, P. G. Kshirsagar , "A text book of Engineering Physics" , S. Chand & Co. Ltd. Revised Edition 2014.

ENVIRONMENTAL SCIENCE AND ENGINEERING

L	T	P	C
2	0	0	2

COURSE OBJECTIVES:

- To find and implement scientific, technological, economic and political solutions to environmental problems
- To study the interrelationship between living organisms and the environment
- To appreciate the importance of the environment by assessing its impact on the human world; envision the surrounding environment, its functions and its value
- To study the dynamic processes and understand the features of the earth‘s interior and surface
- To study the integrated themes and biodiversity, natural resources, pollution control and waste management

ENVIRONMENT AND ECOSYSTEMS: **6**

Definition, scope and importance of Environment - Need for public awareness - Concept of an Ecosystem - Structure and function of an Ecosystem - producers, consumers and decomposers - Energy Flow in the Ecosystem - Ecological succession - Food chains, Food webs and ecological pyramids - Introduction, types, characteristic features, structure and function of the a) Forest Ecosystem, b) Grassland Ecosystem, c) Desert Ecosystem, d) Aquatic Ecosystems (Ponds, Streams, Lakes, Rivers, Oceans, Estuaries)

ENVIRONMENTAL POLLUTION: **6**

Definition - causes, effects and control measures of: a) Air Pollution, b) Water Pollution, c) Soil Pollution, d) Marine Pollution, e) Noise Pollution, f) Thermal Pollution, g) Nuclear Hazards - Soil waste management: causes, effects and control measures of municipal solid wastes - Role of an individual in prevention of pollution - Pollution case studies - Disaster management: floods, earthquake, cyclone and landslides - Field study of local polluted site - urban/rural/industrial/agricultural

NATURAL RESOURCES: **7**

Forest Resources: Use and over-exploitation, deforestation, case studies - Timber Extraction, mining, dams and their effects on forests and tribal people - Water resources: use and over-utilization of surface and ground water, floods, drought, conflicts over water, dams, benefits and problems - Mineral resources: use and exploitation, environmental effects of extracting and using mineral resources, case studies - Food resources: world food problems, changes caused by agriculture and overgrazing, fertilizer-pesticide problems, water logging, salinity, case studies - Energy resources: growing energy needs, renewable and non-renewable energy sources, use of alternate energy sources, case studies - Land resources: land as a resource, land degradation, man induced landslides, soil erosion and desertification - role of an individual in

conservation of natural resources - Equitable use of resources for sustainable lifestyles, field study of local area to document environmental assets - river/forest/grassland/hill/mountain

SOCIAL ISSUES AND THE ENVIRONMENT: 6

From unsustainable to sustainable development - Urban problems related to energy - Water conservation, rain water harvesting, watershed management - Environmental ethics: issues and possible solutions - Climate change, global warming, acid rain, ozone layer depletion, nuclear accidents and holocaust, case studies - Environment Protection Act - Air (prevention and control of pollution) Act - Water (prevention and control of pollution) Act - Wildlife Protection Act - Forest Conservation Act - Enforcement machinery involved in environmental legislation - Central and State Pollution Control Boards - Public awareness

HUMAN POPULATION AND THE ENVIRONMENT: 5

Population growth, variation among nations - Population explosion - Family Welfare Programme - Environment and Human Health - Human Rights - Value Education - HIV/ AIDS - Women and Child Welfare - Role of Information Technology in Environment and Human Health - Case studies

TOTAL PERIODS: 30

COURSE OUTCOMES:

Upon completion of the course, the students will be able to

- Appreciate concepts and methods from ecological and physical sciences and their application in environmental problem solving
- Reflect critically about their roles and identities as citizens, consumers and environmental actors in a complex, interconnected world
- Identify the major sources, effects and monitoring of air and water pollutants
- Appreciate that one can apply systems concepts and methodologies to analyze and understand interactions between social and environmental processes

TEXT BOOKS:

1. Gilbert M. Masters, “Introduction to Environmental Engineering and Science”, Pearson Education , 2nd edition, 2004.
2. Benny Joseph, “Environmental Science and Engineering”, Tata McGraw-Hill, 1st edition, 2017.

PROGRAMMING IN C

L	T	P	C
3	0	0	3

COURSE OBJECTIVES:

To develop C programs using basic programming constructs

- To develop C programs using arrays and strings
- To develop applications in C using functions, pointers and structures
- To perform input/output and file handling

C PROGRAMMING BASICS: **12**

Introduction to C programming: fundamentals - structure of a C program - compilation and linking processes - Constants, Variables, Data Types - Expressions using operators in C - Managing Input and Output operations - Decision Making and Branching - looping statements - solving simple scientific and statistical problems

ARRAYS AND STRINGS: **10**

Arrays - Initialization - Declaration - One dimensional and Two dimensional arrays - Strings: String operations - String Arrays - Simple programs: sorting, searching, matrix operations

FUNCTIONS AND POINTERS: **8**

Function: Definition of function - Declaration of function - Pass by value - Pass by reference - Recursion - Pointers: Definition - Initialization - Pointers arithmetic - Pointers and arrays

STRUCTURES AND UNION: **8**

Introduction - need for Structure data type - Structure definition - Structure declaration - Structure within a structure - Union - Programs using Structures and Unions - Storage classes - Preprocessor directives - Simple programs: singly linked list, doubly linked list

FILE HANDLING AND ADDITIONAL FEATURES IN C: **7**

Console input output functions - disk input output functions - data files - Additional Features in C: command line arguments, bit wise operators, enumerated data types, type-casting

TOTAL PERIODS: 45

COURSE OUTCOMES:

Upon completion of the course, the students will be able to

- Develop simple applications in C using basic constructs
- Design and implement applications using arrays and strings
- Develop and implement applications in C using functions and pointers
- Develop applications in C using structures

TEXT BOOK:

1. Kernighan, B.W and Ritchie, D. M, “The C Programming language”, 2nd edition, Pearson Education, 2006

DIGITAL DESIGN + LAB

L	T	P	C
2	0	2	3

COURSE OBJECTIVES:

- To understand number systems and conversions from one system to another
- To study the components of digital circuits.
- To get an in-depth knowledge of the design of digital circuits.
- To study and design hazard-free circuit.

INTRODUCTION TO DIGITAL SYSTEMS: **9**

Number systems: Decimal, Binary, Octal and Hexadecimal - Conversion from one system to another - Floating point representation of numbers - Arithmetic operations - 1's complement, 2's complement - Study on BCD: Codes - Introduction to Digital Circuits: Advantages and Disadvantages of Digital circuits over Analog circuits - Logic gates: truth tables.

BOOLEAN ALGEBRA AND MINIMIZATION TECHNIQUES: **7**

Introduction to basic laws of Boolean Algebra - Mixed logic: Multilevel gating networks - Sum of products and Product of sum - Simplification of four variable Boolean equations using Karnaugh maps, Quine-McClusky method.

COMBINATIONAL LOGIC CIRCUITS: **7**

Binary Adder and Subtractor: Half adder - Full adder - Half Subtractor - Full Subtractor - 4 bit parallel adder and subtractor - 3-bit binary decoder - Two phase method - Decimal to BCD encoder - 8-to-1 multiplexer - 1-to-8 multiplexer.

SEQUENTIAL LOGIC CIRCUITS: **7**

Flip-flops: Triggering of flip-flops (SR, D, JK and T) - Study of 3 bit and 4 bit binary asynchronous counter - Design of synchronous counter - Shift registers (SISO, SIPO, PISO, PIPO) - Memories (RAM, ROM, EPROM, FLASH) - State Diagram - State Table - State Assignment.

TOTAL PERIODS: 30

SUGGESTIVE LIST OF EXPERIMENTS:

1. Verification of Boolean Theorems using basic gates
2. Design and implementation of combinational circuits using basic gates for arbitrary functions, code converters
3. Design and implement half/full adder and subtractor
4. Design and implement shift registers
5. Design and implement synchronous counters
6. Design and implement asynchronous counters

TOTAL PERIODS: 15

COURSE OUTCOMES:

Upon completion of this course the students will be able to:

- Do conversions among number systems
- Design combinational and sequential logic circuits
- Simplify Boolean functions
- Design using Programmable Logic Devices

TEXT BOOK:

1. M. Morris R. Mano, Michael D. Ciletti, “Digital Design: With an Introduction to the Verilog HDL, VHDL, and System Verilog”, Pearson Education, 6th Edition, 2017.

PROGRAMMING IN C LAB

L	T	P	C
0	0	4	2

COURSE OBJECTIVES:

- To develop programs in C using basic constructs
- To develop applications in C using strings, pointers, functions, structures
- To develop applications in C using file processing

SUGGESTIVE LIST OF EXPERIMENTS:

1. Compiling and Executing C Programs in Linux Environment
2. Programs using I/O Statements and Expressions
3. Programs using Decision making statements
4. Programs using looping statements
5. Programs using 1D Arrays
6. Programs using 2D Arrays
7. Programs using Strings
8. Programs using Functions
9. Programs using Recursion
10. Programs using Pointers
11. Programs using Structures and Unions
12. File handling

TOTAL PERIODS: 30

COURSE OUTCOMES:

Upon completion of the course, the students will be able to

- Develop C programs for simple applications making use of basic constructs, arrays and strings
- Develop C programs involving functions, recursions, pointers, and structures
- Design applications using sequential and random access file processing

ENGINEERING PHYSICS LAB

L	T	P	C
0	0	4	2

COURSE OBJECTIVE:

- To determine the physical, electrical and optical properties of materials

LIST OF EXPERIMENTS:

1. Determination of velocity and compressibility of the given liquid - Ultrasonic interferometer
2. Determination of Planck's Constant
3. Determination of specific resistance of the given wire - Carey Foster Bridge
4. Determination of Energy bandgap of the given semiconductor - Band Gap of Semi-Conductor
5. Determination of grating element / average size of the particles of a given powder sample using laser
6. Determine the numerical aperture, acceptance angle & losses in fibres of the given optical fibre cable

TOTAL PERIODS: 30

COURSE OUTCOMES:

On successful completion of this course, the students will be able to:

- Determine ultrasonic velocity in a medium and associated material properties
- Use principles of dual nature of light to determine universal constants and observe photoelectric effects
- Determine electrical properties of metals and semiconductors like specific resistance of a conductor and bandgap of semiconductors
- Determine Wave length of Semiconductor Lasers or size of grating elements
- Characteristics of Optical Fibres like Numerical Aperture & Acceptance Angle

SEMESTER 2

ENGLISH FOR ENGINEERS

L	T	P	C
2	1	0	3

COURSE OBJECTIVES:

- Develop strategies and skills to enhance their ability to read and comprehend texts in engineering and technical contexts.
- Foster their ability to write convincing job applications and effective reports.
- Develop their speaking skills to make technical presentations, participate in group discussions.
- Strengthen their listening skill which will help them comprehend lectures and talks in their areas of specialization.
- To help learners develop vocabulary, as required in engineering contexts
- To help learners gain the expertise required in grammar for them to function well in engineering contexts

INTRODUCTION: 9

Listening- Listening to product descriptions, talks mostly of a scientific/technical nature and completing information-gap exercises; Speaking -Describing a product; Asking for and giving directions
Reading - Reading descriptions, short technical texts from journals- newspapers
Writing- purpose statements - extended definitions - writing instructions - checklists- recommendations; note-making and note-taking
Vocabulary Development- technical vocabulary, avoiding jargon
Language Development -subject verb agreement - compound words.

READING AND WRITING TECHNICAL TEXTS: 9

Listening- Listening to longer technical talks and completing exercises based on them
Speaking- describing a process; making enquiries
Reading - reading longer technical texts- identifying the various transitions in a text- paragraphing Writing- interpreting charts, graphs; writing formal letters/emails including complaints
Vocabulary Development- vocabulary used in formal letters/emails and reports
Language Development- impersonal passive voice, numerical adjectives.

BECOMING INDEPENDENT USERS OF LANGUAGE FOR TECHNICAL CONTEXTS: 9

Listening- Listening to classroom lectures/ talks on engineering/technology
Speaking - introduction to technical presentations
Reading - longer texts both general and technical, practice in speed reading
Writing-Describing a process, use of sequence words; compare and contrast paragraphs
Vocabulary Development- sequence words- Misspelled words
Language Development- embedded sentences

LANGUAGE FOR JOB-PREPAREDNESS: **9**

Listening- Listening to documentaries and making notes
Speaking - mechanics of presentations
Reading - reading for detailed comprehension
Writing- email etiquette - job application - cover letter -Résumé preparation (via email and hard copy)- analytical essays and issue-based essays
Vocabulary Development- finding suitable synonyms-paraphrasing
Language Development- clauses- if conditionals

ADVANCED READING AND WRITING: **9**

Listening- TED/Ink talks
Speaking -participating in a group discussion
Reading- reading and understanding technical articles
Writing- Writing reports- minutes of a meeting- accident and survey
Vocabulary Development- verbal analogies
Language Development- reported speech

TOTAL PERIODS: 45

COURSE OUTCOMES:

On successful completion of this course, the learners will be able to

- Read technical texts and write area- specific texts effortlessly.
- Listen and comprehend lectures and talks in their area of specialization successfully.
- Speak appropriately and effectively in varied formal and informal contexts.
- Write reports and winning job applications.

TEXT BOOK:

1. Sudarshana, N.P. and Savitha, C. (2018). English for Engineers. Cambridge University Press.

STATISTICAL FOUNDATIONS OF DATA SCIENCE

L	T	P	C
3	0	0	3

COURSE OBJECTIVES:

- To provide the fundamental concepts of probability and random variable
- To learn different statistical methods needed for data analysis
- To introduce some standard distributions applicable to engineering
- To understand the basic concepts in two dimensional random variables
- To understand the basic concepts of random processes which are widely used in IT fields

PROBABILITY THEORY: **9**

Probability - axioms - laws of probability - total probability - Bayes' Theorem - random variables - distribution functions - mass and density functions

STATISTICAL AVERAGES: **9**

Mathematical expectation of a random variable - properties of expectation - median - mode - variance - Kurtosis - skewness - moments - moment generating function

PROBABILITY DISTRIBUTIONS: **9**

Bernoulli - Binomial - Poisson - Multinomial - Uniform - Exponential and Gaussian distributions - Central Limit Theorem (for independent and identically distributed random variables)

TWO-DIMENSIONAL RANDOM VARIABLES: **9**

Joint distribution - marginal distribution - conditional distribution - joint density function - marginal density function - conditional density function - covariance - correlation and regression lines

RANDOM PROCESSES: **9**

Definition - Classification - Stationary Process - Markov Processes and Markov chain

TOTAL PERIODS: 45

COURSE OUTCOMES:

After the completion of this course, students will be able to:

- Understand the fundamental concepts of probability and random variable
- Apply the various statistical methods needed to analyze the given data
- Have knowledge of standard distributions which can describe real life phenomenon.
- Understand the basic concepts two dimensional random variables and apply in engineering applications
- Apply the concept of random processes in data analysis

TEXT BOOK:

1. Gupta, S.C. and Kapoor, V.K. (2007): Fundamentals of Mathematical Statistics, 11th Edn., (Reprint), Sultan Chand and Sons.

PROGRAMMING IN PYTHON

L	T	P	C
2	0	0	2

COURSE OBJECTIVES:

- To solve algorithmic problems
- To compose programs in Python using iteration and recursion
- To construct programs in Python using functions
- To handle file operations using Python

DATA, EXPRESSIONS, STATEMENTS, CONDITIONALS: 8

Data and types: int, float, boolean, string, list - variables - expressions - statements - simultaneous assignment - precedence of operators - comments - in-built modules and functions - Conditional: boolean values and operators, conditional (if), alternative (if-else), case analysis (if-elif-else)

ITERATION, FUNCTIONS, STRINGS: 8

Iteration: while, for, break, continue, pass - Functions: function definition, function call, flow of execution, parameters and arguments, return values, local and global scope, recursion - Strings: string slices, immutability, string functions and methods, string module

CONTAINERS: 8

Lists: list operations, list slices, list methods, list loop, mutability, aliasing, cloning lists, list parameters, nested lists, list comprehension - Dictionaries: operations and methods, looping and dictionaries, reverse lookup, dictionaries and lists, dictionary comprehension - Tuples: tuple assignment, tuple as return value, tuple operations

FILES AND EXCEPTION HANDLING: 6

Files: Text files, reading and writing files, format operator, file names and paths - command line arguments - Exceptions: try-catch, types of exception handling

TOTAL PERIODS: 30

COURSE OUTCOMES:

After the completion of this course, students will be able to:

- Think logically to solve programming problems using Python
- Understand and develop simple Python programs using conditionals and loops
- Decompose a program into functions
- Represent compound data using Python lists, tuples, dictionaries
- Perform input/output with files

TEXT BOOK:

1. Allen B. Downey, “Think Python: How to Think Like a Computer Scientist”, Shroff/O'Reilly; 2nd edition, 2016.

DATA STRUCTURES

L	T	P	C
3	0	0	3

COURSE OBJECTIVES:

- To understand the concepts of ADTs
- To learn linear data structures - lists, stacks, and queues
- To understand the various sorting and searching algorithms
- To gain knowledge on tree and graph data structures
- To study the concepts and algorithms in hashing

INTRODUCTION TO DATA STRUCTURES: **4**

Basic Terminology - Data Organization - Abstract Data Types - Data Structures: Types and Operations - Time and Space Complexity analysis: Big-Oh, Ω, θ, little-oh notations - Growth rates - Time-Space Tradeoff - Time complexity analysis of some example problems.

LISTS: **7**

List ADT - Array Implementation of List - Operations on lists: Insertion, Deletion, Merging - Linked Lists: Singly Linked list, Doubly linked list, Circular linked list - Operations on linked lists - The Polynomial ADT - Cursor implementation of lists

STACKS: **6**

Stack ADT: Array Implementation, Linked list implementation - Operations on Stacks - Applications of stacks: Balancing Symbols, Postfix expression evaluation, Infix to postfix conversion - Function calls - Recursion.

QUEUES: **5**

Queue ADT: Array Implementation, Linked list implementation - Operations on Queues - Circular Queue - Double ended queue - Priority Queue - Applications of Queue.

TREES: **7**

Tree ADT - Implementation of trees - Tree traversals - Binary trees - Binary Search Trees (BST): Operations on BSTs - Expression trees - AVL trees: Operations on AVL trees - Splay trees - Red-Black trees - B-Trees - Heaps - Types of heaps.

GRAPHS: **7**

Graph ADT - Implementation of Graphs - Traversal: Breadth First Search, Depth first search - Topological sort - Minimum spanning tree: Prim's Algorithm & Kruskal's algorithm - Dijkstra's Algorithm - Applications of graphs.

SORTING AND SEARCHING: **6**

Searching: Linear Search, Binary Search - Sorting: Bubble sort, Selection sort, Insertion sort, Quick Sort, Merge Sort, Shell sort, Counting Sort.

HASHING: **3**

Hash Tables - Hash Functions - Separate Chaining - Linear Probing - Quadratic Probing - Open addressing - Rehashing - Extendible hashing.

TOTAL PERIODS: 45

COURSE OUTCOMES:

Upon completion of the course, the students will be able to

- Describe the importance of data structures in programming
- Explain the concepts and applications of various linear data structures
- Identify the suitable non-linear data structure for a given context
- Use appropriate searching and sorting algorithms for a problem
- Describe the concepts and algorithms related to hashing

TEXT BOOK:

1. Mark Allen Weiss, “Data Structures & Algorithm Analysis in C”, Pearson Education, Second Edition, 2002.

COMPUTER ORGANIZATION

L	T	P	C
3	0	0	3

COURSE OBJECTIVES:

- To make students understand the basic structure and operation of digital computer
- To understand the hardware-software interface
- To familiarize the students with arithmetic and logic unit and implementation of fixed point and floating-point arithmetic operations
- To expose the students to the concept of pipelining
- To familiarize the students with hierarchical memory system including cache memories and virtual memory
- To expose the students with different ways of communicating with I/O devices and standard I/O interfaces

PROCESSOR FUNDAMENTALS: **9**

Computer Components - Performance Metrics - Instruction set architecture - Various addressing modes - Instruction execution in ALU - Simple data path

COMPUTER ARITHMETIC: **12**

Representing unsigned and signed integer numbers - Floating point system - Integer addition and subtraction - Adders: Ripple carry adder, Carry Look Ahead adders - Integer multiplication and division - High-Radix Multipliers and High-Radix Dividers - Redundant number systems - Residue number systems

MEMORY SYSTEMS: **9**

Memory hierarchy - Cache Memory: Organization, Design - Virtual Memory concepts

INTERCONNECTIONS AND PERIPHERALS: **6**

Interconnection structures, Bus - PCI, Mesh, Hyper cube, Ring, Star - I/O Interface Systems: Keyboard, Monitor, Mouse, Bluetooth, USB, Flash

ILP ARCHITECTURES: **9**

Pipelining - Hazards in pipelining - Super pipelining - Super scalar - VLIW - Combining super scalar and VLIW with pipelining

TOTAL PERIODS: 45

COURSE OUTCOMES:

On successful completion of this course, the students will be able to

- Explain processor fundamentals
- Design arithmetic and logic unit
- Evaluate performance of memory systems
- Extend the learning to parallel processing architectures
- Explain interconnection structures

TEXT BOOK:

1. Carl Hamacher, Zvonko Vranesic, Safwat Zaky, “Computer Organization”, McGraw Hill Education, Fifth Edition, 2011.

FOUNDATIONS OF DATA SCIENCE + LAB

L	T	P	C
3	0	2	4

COURSE OBJECTIVES:

- To learn fundamentals of Data Science using Python
- To understand probability distributions and statistical Inferences
- To be familiar with supervised and unsupervised methods in machine learning
- To explore the algorithms used for analysing massive data problems and social networks
- To learn about visualization

INTRODUCTION TO DATA SCIENCE: **9**

Introduction: Need for data science - Benefits and uses - Facets of data - Big data ecosystem - The data science process: Retrieving data - Cleansing, integrating and transforming data - Data analysis - Build the models - Presenting findings and building applications

TOOLBOXES FOR DATA SCIENTISTS: **10**

Introduction to Python - Fundamental Python Libraries for Data Scientists: Numpy - Scipy - ScikitLearn - Pandas - Matplotlib - IDE - Data Manipulation with Python

DESCRIPTIVE STATISTICS: **8**

Introduction - Data Preparation - Exploratory Data Analysis: Data summarization - Data distribution - Outlier Treatment - Measuring asymmetry - Continuous distribution; Estimation: Mean - Variance - Sampling - Covariance - Correlation

STATISTICAL INFERENCE: **8**

Introduction - Frequentist Approach - Measuring the Variability in Estimates: Point estimates - Confidence intervals; Hypothesis Testing: Using confidence intervals - Using p-values

MACHINE LEARNING: **10**

Supervised Learning: Introduction - kNN classifier; Regression analysis: Linear regression - Logistic regression; Unsupervised Learning: Introduction - Clustering - Evaluation metrics

TOTAL PERIODS: 45

SUGGESTIVE LIST OF EXERCISES:

1. Basic python programming constructs
2. Python Lists, tuples and dictionary programs
3. Introduction to Jupyter Notebook
4. Introduction to Numpy and pandas
5. Handling dataframes using pandas
6. Introduction to statistics in python
7. Exercises with SciPy

TOTAL PERIODS: 15

COURSE OUTCOMES:

After the completion of this course, students will be able to:

- Develop Python programs to perform analysis on data
- Understand various probability distributions and statistical inferences
- Develop applications to demonstrate machine learning algorithms in practice
- Understand the principles of handling data streams
- Discuss topic and graphical modeling techniques in real world problem

TEXT BOOK:

1. Davy Cielen, Arno D B Meysman, Mohamed Ali, “Introducing Data Science - Big data, Machine Learning, and more using Python tools”, Manning Publications Co, 2016.

DATA STRUCTURES LAB

L	T	P	C
0	0	4	2

COURSE OBJECTIVES:

- To develop programs in C for implementation of various data structures
- To design and implement linear data structures - lists, stacks, and queues
- To implement sorting, searching and hashing algorithms
- To solve problems using tree and graph structures

SUGGESTIVE LIST OF EXERCISES:

1. Array Implementation of Lists
2. Linked List Implementation of lists
3. Doubly and Circular linked lists
4. Implementation of Stack Data Structure (Array and linked list implementations)
5. Conversion of infix to postfix expression and evaluation of postfix expression
6. Implementation of Queue Data Structure (Array and linked list implementations)
7. Implementation of Circular and Double ended queue
8. Implementation of Binary Search tree - insertion, deletion & traversals
9. Implementation of Graphs - BFS & DFS
10. Finding Minimum Spanning Tree for a graph
11. Implementation of Hash table
12. Implementation of Linear Search & Binary search
13. Implementation of Bubble sort, insertion sort and selection sort
14. Implementation of Quick sort and merge sort
15. Solve a problem using appropriate data structures

TOTAL PERIODS: 30

COURSE OUTCOMES:

Upon completion of the course, the students will be able to

- Develop programs using appropriate linear data structures.
- Identify and implement the suitable non-linear data structure for a given problem.
- Implement appropriate searching and sorting algorithms for a problem.
- Develop programs using basic hashing concepts.

PROGRAMMING IN PYTHON LAB

L	T	P	C
0	0	4	2

COURSE OBJECTIVES:

- To solve problems using algorithms and flowcharts
- To write, test, and debug simple Python programs
- To develop and execute programs using Python programming constructs

SUGGESTIVE LIST OF EXERCISES:

1. Use Linux shell commands, use Python in interactive mode, and an editor
2. Write simple programs (area of a geometric shape, simple interest, solve quadratic equation, net salary)
3. Write programs using conditional statements (leap year, maximum of 2 numbers, maximum of 3 numbers, simple calculator, grade of the total mark)
4. Develop programs using loops and nested loops (gcd, prime number, integer division, sum of digits of an integer, multiplication table, sum of a series, print patterns, square root using Newton's method)
5. Develop programs using functions (sine and cosine series, Pythagorean triplets)
6. Develop programs using recursion (efficient power of a number, factorial, Fibonacci number)
7. Develop programs using strings (palindrome, finding substring) without using in-built functions
8. Develop programs using lists and tuples (linear search, binary search, selection sort, insertion sort, quicksort)
9. Develop programs using nested lists (matrix manipulations)
10. Develop simple programs using dictionaries (frequency histogram, nested dictionary)
11. Develop programs using Files (read and write files)
12. Develop programs to perform any task by reading arguments from the command line

TOTAL PERIODS: 30

COURSE OUTCOMES:

After the completion of this course, students will be able to:

- To write, test, and debug simple Python programs
- To implement Python programs with conditionals and loops
- Use functions for structuring Python programs
- Represent compound data using Python lists, tuples, and dictionaries
- Read and write data from/to files in Python