



DEPARTMENT
OF
COMPUER SCIENCE AND ENGINEERING

RM21 REGULATION

B.Tech- COURSE STRUCTURE AND SYLLABUS

(I YEAR AND II YEAR)

COURSE STRUCTURE

I Year – I SEMESTER						
S.NO	SUBJECT CODE	SUBJECT NAME	L	T	P	CREDITS
1	21HS1101	Communicative English	3	0	0	3
2	21BS1102	Linear Algebra & Differential Equations	3	0	0	3
3	21BS1105	Applied Chemistry	3	0	0	3
4	21ES1108	Programming for Problem Solving using C	3	0	0	3
5	21ES1109	Engineering Graphics	3	0	0	3
6	21HS1151	English Communication Skills Lab	0	0	3	1.5
7	21BS1156	Applied Chemistry Lab	0	0	3	1.5
8	21ES1158	C Programming Lab	0	0	3	1.5
9	21MC1101	Constitution of India	2	0	0	0
						19.5

I Year – II SEMESTER						
S.NO	SUBJECT CODE	SUBJECT NAME	L	T	P	CREDITS
1	21BS1201	Numerical Methods & Vector Calculus	3	0	0	3
2	21BS1203	Applied Physics	3	0	0	3
3	21ES1205	Basic Electrical Engineering	3	0	0	3
4	21ES120A	Digital Logic Design	3	0	0	3
5	21ES121C	Python Programming	3	0	0	3
6	21BS1252	Applied Physics Lab	0	0	3	1.5
7	21BS1253	Applied Physics Virtual Lab	0	0	2	0
8	21ES1255	Basic Electrical Engineering Lab	0	0	3	1.5
9	21ES1258	Python Programming Lab	0	0	3	1.5
10	21MC1201	Environmental Studies	2	0	0	0
Total Credits						19.5

II Year – I SEMESTER						
S.NO	SUBJECT CODE	SUBJECT NAME	L	T	P	CREDITS
1	21BS2101	Transform Techniques	3	0	0	3
2	21BS2102	Mathematical Foundations of Computer Science	3	0	0	3
3	21CS2101	Operating Systems	3	0	0	3
4	21CS2102	Object Oriented Programming through C++	3	0	0	3
5	21CS2103	Software Engineering	3	0	0	3
6	21CS2152	Operating Systems Lab	0	0	3	1.5
7	21CS2153	Object Oriented Programming through C++ Lab	0	0	3	1.5
8	21CS2154	Software Engineering Lab	0	0	3	1.5
9	21CS2155	Applications of Python-NumPy	0	0	4	2
10	21MC2101	Essence of Indian Tradition Knowledge	2	0	0	0
Total Credits						21.5

II Year – II SEMESTER						
S. No	Course Code	Courses	L	T	P	Credits
1	BS	Probability and Statistics	3	0	0	3
2	CS	Database Management Systems	3	0	0	3
3	CS	Formal Languages and Automata Theory	3	0	0	3
4	CS	Java Programming	3	0	0	3
5	HS	Managerial Economics and Financial Accountancy	3	0	0	3
6	CS	Database Management Systems Lab	0	0	2	1
7	CS	R Programming Lab	0	1	2	2
8	CS	Java Programming Lab	0	0	3	1.5
9	SO	Skill Oriented Course - II Applications of Python-Pandas	0	0	4	2
Total Credits			21.5			

CSE III B. Tech – I Semester						
S.No	Course Code	Courses	Hours per week			Credits
			L	T	P	
1	PC	Computer Networks	3	0	0	3
2	PC	Design and Analysis of Algorithms	3	0	0	3
3	PC	Data Warehousing and Data Mining	3	0	0	3
4	Open Elective / Job Oriented	Open Elective-I 1.Disaster Management 2.Environmental Pollution 3.Traffic Safty	3	0	0	3
5	PE	Professional Elective-I 1. Artificial Intelligence 2. Compiler Design 3.Distributed Systems	3	0	0	3
6	PC	Data Warehousing and Data Mining Lab	0	0	3	1.5
7	PC	Computer Networks Lab	0	0	3	1.5
8	SO	Skill Oriented Course – III 1.Dev Ops	0	0	4	2
9	MC	Employability Skills-I	2	0	0	0
10	PR	Summer Internship 2 Months (Mandatory) after second year (to be evaluated during V semester	0	0	0	1.5
Total credits						21.5

CSE III B. Tech – II Semester						
S.No	Course Code	Courses	Hours per week			Credits
			L	T	P	
1	PC	Big Data Analytics	3	0	0	3
2	PC	Cloud Computing	3	0	0	3
3	PC	Cryptography and Network Security	3	0	0	3
4	PE	Professional Elective-II 1.Mobile Computing 2.Software project management 3.Object Oriented Analysis andDesign	3	0	0	3
5	Open Elective /Job Oriented	Open Elective-II IOT	3	0	0	3
6	PC	Big Data Analytics Lab	0	0	3	1.5
7	PC	Cloud Computing Lab	0	0	3	1.5
8	PC	Cryptography and Network Security Lab	0	0	3	1.5
9	SO	Skill Oriented Course - IV MEAN Stack Technologies-Module I	0	0	4	2
10	MC	Employability skills-II	2	0	0	0
Total credits						21.5

CSE IV B. Tech –I Semester						
S.No	Course Code	Course Title	Hoursperweek			Credits
			L	T	P	C
1	PE	Professional Elective-III 1.Fog Computing 2.Neural Networks and Soft Computing 3.Ad-hoc and Sensor Networks 4.Cyber Security & Forensics	3	0	0	3
2	PE	Professional Elective-IV 1. Deep Learning Techniques 2. Social Networks & Semantic Web 3. Computer Vision 4.MOOCs-NPTEL/SWAYAM	3	0	0	3
3	PE	Professional Elective-V 1.Block-Chain Technologies 2.Wireless Network Security 3.Ethical Hacking 4.MOOCs-NPTEL/SWAYAM ⁹⁶	3	0	0	3
4	Open Elective /Job Oriented	Open Elective-III 1.Computer Graphics 2.Software Testing Methodologies 3.Expert Systems	3	0	0	3
5	SO	1. Robotic Process Automation OR 2.MEAN Stack Technologies-Module II- Angular JS and MongoDB	0	0	4	2
7	PR	Industrial/Research Internship 2 months (Mandatory) after third year (to be evaluated during VII semester	0	0	0	3
Total credits						17

CSE IV B. Tech –II Semester						
S.No	Course Code	Course Title	Hours per week			Credits
			L	T	P	C
1	Project	Major Project Work, Seminar Internship	-	-	-	12
Total credits						12



Evaluation-Distribution and Weightage of marks

THEORY COURSES

S.No	Component	Internal	External	Total
1.	Theory	30	70	100

Internal marks can be calculated with 80% weightage for better of the two mids and 20% Weightage for another mid exam.

Example:

Mid-1 marks = Marks secured in (descriptive examination-1 + one assignment-1)

Mid-2 marks = Marks secured in (descriptive examination-2 + one assignment-2)

Final internal Marks =

(Best of (Mid-1/Mid-2) marks x 0.8 + Least of (Mid-1/Mid-2) marks x 0.2)

LABORATORY COURSES

S.No	Component	Internal	External	Total
1.	Practical	15	35	50

Internal Evaluation		
S.No	Split up	Marks
1	Day-to-day work	5
2	Record	5
3	Internal test	5
Total		15

External Evaluation		
S.No	Split up	Marks
1	Procedure/Algorithm	10
2	Experimentation / Program execution	10
3	Result/Inference	5
4	Viva voce	10
5	Total	35

For the subject having design and / or drawing (such as Engineering Graphics, Engineering Drawing, Machine Drawing) and estimation, the distribution shall be 30 marks for internal evaluation (15 marks for continuous Assessment (day-to-day work) and 15 marks for internal tests) and 70 marks for end examination. There shall be two internal tests in a Semester for 15 marks each and final marks can be calculated with 80% weightage for better of the two tests and 20% weightage for other test and these are to be added to the marks obtained in day-to-day work.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

COMMUNICATIVE ENGLISH

Introduction

The course is designed to train students in receptive (listening and reading) as well as productive and interactive (speaking and writing) skills by incorporating a comprehensive, coherent and integrated approach that improves the learners' ability to effectively use English language in academic/workplace contexts. The shift is from learning about the language to using the language. On successful completion of the compulsory English language course/s in B.Tech., learners would be confident of appearing for international language qualification/proficiency tests such as IELTS, TOEFL, or BEC, besides being able to express themselves clearly in speech and competently handle the writing tasks and verbal ability component of campus placement tests. Activity based teaching-learning methods would be adopted to ensure that learners would engage in actual use of language both in the classroom and laboratory sessions.

Course Objectives:

- Facilitate effective listening skills for better comprehension of academic lectures and English spoken by native speakers
- Focus on appropriate reading strategies for comprehension of various academic texts and authentic materials
- Help improve speaking skills through participation in activities such as role plays, discussions and structured talks/oral presentations
- Impart effective strategies for good writing and demonstrate the same in summarizing, writing well organized essays, record and report useful information
- Provide knowledge of grammatical structures and vocabulary and encourage their appropriate use in speech and writing
- To develop and integrate the use of the four language skills i.e. Reading, Listening, Speaking.

Course Outcomes:

At the end of the module, the learners will be able to

- Understand social or transactional dialogues spoken by native speakers of English and identify the context, topic, and pieces of specific information
- Employ suitable strategies for skimming and scanning to get the general idea of a text and locate specific information
- Recognize paragraph structure and be able to match beginnings/endings/headings with paragraphs
- Ask and answer general questions on familiar topics and introduce oneself/others
- Form sentences using proper grammatical structures and correct word forms.
- Develop your public speaking abilities by giving you opportunities to speak in class, both informally and formally.

Unit 1:

Lesson-1: A Drawer full of happiness from “**Infotech English**”, Maruthi Publications

Lesson-2: Deliverance by Premchand from “**The Individual Society**”, Pearson Publications. (Non-detailed)

Listening: Listening to short audio texts and identifying the topic. Listening to prose, prose and conversation.

Speaking: Asking and answering general questions on familiar topics such as home, family, work, studies and interests. Self introductions and introducing others.

Reading: Skimming text to get the main idea. Scanning to look for specific pieces of information.

Reading for Writing: Paragraph writing (specific topics) using suitable cohesive devices; linkers, sign posts and transition signals; mechanics of writing - punctuation, capital letters.

Vocabulary: Technical vocabulary from across technical branches (20) GRE Vocabulary (20) (Antonyms and Synonyms, Word applications) Verbal reasoning and sequencing of words.

Grammar: Content words and function words; word forms: verbs, nouns, adjectives and adverbs; nouns: countables and uncountables; singular and plural basic sentence structures; simple question form - wh-questions; word order in sentences.

Unit 2:

Lesson-1: Nehru's letter to his daughter Indira on her birthday from “**Infotech English**”, Maruthi Publications

Lesson-2: Bosom Friend by Hira Bansode from “**The Individual Society**”, Pearson Publications.(Non-detailed)

Listening: Answering a series of questions about main idea and supporting ideas after listening to audio texts, both in speaking and writing.

Speaking: Discussion in pairs/ small groups on specific topics followed by short structured talks. Functional English: Greetings and leave takings.**Reading:** Identifying sequence of ideas; recognizing verbal techniques that help to link the ideas in a paragraph together.

Reading for Writing: Summarizing - identifying main idea/s and rephrasing what is read; avoiding redundancies and repetitions.

Vocabulary: Technical vocabulary from across technical branches (20 words). GRE Vocabulary Analogies (20 words) (Antonyms and Synonyms, Word applications)

Grammar: Use of articles and zero article; prepositions.

Unit 3:

Lesson-1: Stephen Hawking-Positivity 'Benchmark' from “**Infotech English**”, Maruthi Publications

Lesson-2: Shakespeare's Sister by Virginia Woolf from “**The Individual Society**”, Pearson Publications.(Non-detailed)

Listening: Listening for global comprehension and summarizing what is listened to, both in speaking and writing.

Speaking: Discussing specific topics in pairs or small groups and reporting what is discussed. Functional English: Complaining and Apologizing. R-20 Syllabus for CSE, JNTUK w. e. f. 2020 –**Reading:** Reading a text in detail by making basic inferences - recognizing and interpreting specific context clues; strategies to use text clues for comprehension. Critical reading.

Reading for Writing: Summarizing - identifying main idea/s and rephrasing what is read; avoiding redundancies and repetitions. Letter writing-types, format and principles of letter writing.E-mail etiquette, Writing CV's.

Vocabulary: Technical vocabulary from across technical branches (20 words). GRE Vocabulary (20 words) (Antonyms and Synonyms, Word applications) Association, sequencing of words

Grammar: Verbs - tenses; subject-verb agreement; direct and indirect speech, reporting verbs for academic purposes.

Unit 4:

Lesson-1: Liking a Tree, Unbowed: Wangari Maathai-biography from “**Infotech English**”, Maruthi Publications

Lesson-2: Telephone Conversation-Wole Soyinka from “**The Individual Society**”, Pearson Publications.(Non-detailed)

Listening: Making predictions while listening to conversations/ transactional dialogues without video (only audio); listening to audio-visual texts.

Speaking: Role plays for practice of conversational English in academic contexts (formal and informal) - asking for and giving information/directions. Functional English: Permissions, Requesting, Inviting.

Reading: Studying the use of graphic elements in texts to convey information, reveal trends/patterns/relationships, communicative process or display complicated data.

Unit 5:

Lesson-1: Stay Hungry-Stay foolish from “**Infotech English**”, Maruthi Publications

Lesson-2: Still I Rise by Maya Angelou from “**The Individual Society**”, Pearson Publications.(Non-detailed)

Listening: Identifying key terms, understanding concepts and interpreting the concepts both in speaking and writing.

Speaking: Formal oral presentations on topics from academic contexts - without the use of PPT slides. Functional English: Suggesting/Opinion giving.

Reading: Reading for comprehension. RAP Strategy Intensive reading and Extensive reading techniques.

UNIT 6:

Functional Grammar

- Close encounter, fill in the blanks of adjectives,
- Degrees of comparison, Dialogues,
- Non-verbal information, Statement of purpose,
- Reported speech, correction of sentences, Reading comprehension,
- Academic proposal (career building essential) Poster presentation.

Text Books:

1. “**Infotech English**”, Maruthi Publications. (Detailed)

2. “**The Individual Society**”, Pearson Publications.(Non-detailed)

Prescribed text book for Laboratory for Semesters-I & II:

1. “**Infotech English**”, Maruthi Publications. (with Compact Disc)

Reference Books

1. Bailey, Stephen. Academic writing: A handbook for international students. Routledge, 2014.
2. Chase, Becky Tarver. Pathways: Listening, Speaking and Critical Thinking. Heinley ELT; 2nd Edition, 2018.
3. Skillful Level 2 Reading & Writing Student's Book Pack (B1) Macmillan Educational.
4. Hewings, Martin. Cambridge Academic English (B2). CUP,

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

LINEAR ALGEBRA & DIFFERENTIAL EQUATIONS

Course objectives:

- To instruct the concept of Matrices in solving simultaneous linear systems.
- Understanding of formation and solution of ordinary differential equations.
- Knowing the mathematical methods to solve applications of differential equations.
- To familiarize a variety of well-known sequences and series, with a developing intuition about the behaviour of new ones.

Course outcomes : At the end of the course, the student will learn

CO1: Apply the knowledge to solve linear system of equations.

CO2: Illustrate the methods of computing Eigen values and Eigen vectors

CO3: Able to solve the differential equations related to various engineering fields

CO4: Determine the solutions of linear differential equations of higher order.

CO5: Familiarize with functions of several variables which is useful in optimization

CO6: Utilize mean value theorems to real life problems

UNIT I: Linear systems of equations

Rank of a matrix by Echelon form and Normal form, PAQ in normal form, Solving system of homogeneous and non-homogeneous linear equations, Gauss Elimination method, Gauss Jordan method.

UNIT II: Eigen values and Eigen vectors

Eigen values and Eigen vectors and properties (without proof), Cayley-Hamilton theorem (without proof)

Applications – Finding the inverse and power of a matrix by Cayley-Hamilton theorem, Reduction to

Diagonal form, Quadratic forms and nature of the quadratic forms, Reduction of quadratic form to canonical forms by orthogonal transformation.

UNIT III: Differential equations of first order and first degree

Linear differential equations, Bernoulli's equations, Exact equations and equations reducible to exact form.

Applications: Newton's Law of cooling – Law of natural growth and decay – Orthogonal trajectories.

UNIT IV: Linear differential equations of higher order

Homogeneous and Non-homogeneous differential equations of higher order with constant coefficients, with non-homogeneous term of the type e^{ax} , $\sin ax$, $\cos ax$, polynomials in x^n , $e^{ax}V(x)$ and $x^nV(x)$, Method of Variation of parameters.

UNIT V: Partial differentiation

Introduction , Homogeneous function , Euler's theorem, Total derivative, Chain rule, Jacobian , Functional dependence , Taylor's and MacLaurin's series expansion of functions of two variables.

Applications: Maxima and Minima of functions of two variables without constraints and Lagrange's method.

UNIT VI: Sequences, Series and Mean value theorems

Convergences and divergence , Ratio test , Comparison tests ,Integral test, Cauchy's root test , Alternate series, Leibnitz's rule.

Mean Value Theorems (without proofs): Rolle's Theorem . Lagrange's mean value theorem , Cauchy's mean value theorem , Taylor's and Maclaurin's theorems with remainders, Problems and applications on the above theorem.

Text Books:

1. **B. S. Grewal**, Higher Engineering Mathematics, 43rd Edition, Khanna Publishers.
2. **B. V. Ramana**, Higher Engineering Mathematics, 2007 Edition, Tata Mc. Graw Hill Education.

Reference Books:

1. **Erwin Kreyszig**, Advanced Engineering Mathematics, 10th Edition, Wiley-India
2. **N.P.BALI &Dr.MANISH GOYAL**, A Text book of Engineering Mathematics, Lakshmi Publications, 9th Edition, 2014.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

APPLIED CHEMISTRY

Course Objectives:

* Plastics are now a days used in household appliances; They are also used as composites (FRP) in aerospace and automotive industries.

* The basics for the construction of galvanic cells are introduced to have understanding on the concepts. Understanding on the concept of Corrosion and Mechanism of Corrosion with Theories like Electrochemical theory.

* With the increase in demand for power and also with depleting sources of fossil fuels, the demand for alternative sources of fuels is increasing. Some of the prospective fuel sources are introduced and the study can create a better understanding on the Non –Conventional Energy Sources.

*Outline the basics of Green chemistry & Molecular switches.

Course Outcomes:

CO1: Evaluate various polymer synthetic methods for applications in Engineering and technological materials.

CO2: Utilize the theory of construction of electrodes, batteries and fuel cells in redesigning new engineering products and categorize the reasons for Corrosion and study methods to control corrosion

CO3: Synthesize nonmaterials for modern advances of engineering technology.
Analyze the applications of liquid crystals and superconductors.

CO4: Analyze the principles of different analytical instruments and their Applications.

CO5: Obtain the knowledge of green chemistry and molecular machines

CO6: Familiarize various renewable energy sources that could be useful to alternate fossil fuels. Design models for energy by different natural sources.

UNIT I: HIGH POLYMERS AND PLASTICS:

Polymerization:- Introduction-Types of polymerization(addition and condensation) - Examples –Physical and mechanical properties –

Plastics :- Thermoplastics and Thermosetting plastics – Compounding and fabrication (compression, injection, blown film and extrusion) - Preparation, properties and applications of PVC, Bakelite and polycarbonates.

Elastomers:- Natural rubber-Draw backs- vulcanization – Synthetic rubbers: preparation, properties and applications of Buna S, thiokol and polyurethanes .

Composite Materials:Fibre reinforced plastics-aramid fibre reinforced plastics.

Conducting polymers-biodegradable polymers.

UNIT II: ELECTROCHEMICAL CELLS AND CORROSION

Galvanic cells :- Single electrode potential – Electro chemical series and uses of this series- Types of Electrodes (Reference electrode) Hydrogen and Calomel electrodes - Batteries: Dry Cell,, Lithium ion and zinc air cell – **Fuel cells**-Hydrogen-oxygen fuel cell and methanol – oxygen fuel cell.

Corrosion:- Definition – Theories of Corrosion (chemical & electrochemical) – Factors which influence the rate of corrosion - Protective coatings:– Electroplating, Electro less plating- Organic surface Coatings-Paints.

UNIT-III: Material Chemistry

Super conductors:-Type –I, Type II-characteristics and applications

Nano materials: Introduction, sol-gel method, characterization by (Brunauer Emmet Teller [BET]), (scanning electron microscopy [SEM]) and (transmission electron microscopy [TEM]), applications of graphene and fullerenes, carbon nanotubes (types, preparation and applications)

Liquid crystals: Introduction-types-applications.

UNIT-IV: Spectroscopy

UV spectroscopy- Basic principle-Instrumentation-Applications

IR spectroscopy- Basic principle-Instrumentation-Applications

NMR spectroscopy- Basic principle-Instrumentation-Applications

UNIT V: CHEMISTRY OF ADVANCED MATERIALS

PART-I:GREEN CHEMISTRY:-Principles of Green chemistry- Methods for Green Synthesis- Aqueous Phase method-Phase transfer catalysis method -Applications of Green Synthesis– R₄M₄ Principles

PART-II: Molecular switches:

Characteristics of molecular motors and machines,

Rotaxanes and Catenanes as artificial molecular machines, prototypes – linear motions in rotaxanes.

UNIT VI: RENEWABLE ENERGY SOURCES

Introduction, types of Renewable energy sources, solar energy – photovoltaic cell: design, working and its Importance. Hydropower includes setup a hydropower plant (schematic diagram)-Geothermal energy: Introduction-schematic diagram of a geothermal power plant-Tidal and wave power- Biomass and bio fuels.

Text Books:

1. P.C. Jain and M. Jain “Engineering Chemistry”, 15/e, Dhanpat Rai & Sons, Delhi, (Latest edition).
2. Shikha Agarwal, “Engineering Chemistry”, Cambridge University Press, New Delhi, (2019).
3. S.S. Dara, “A Textbook of Engineering Chemistry”, S.Chand & Co, (2010).

4. Shashi Chawla, “Engineering Chemistry”, Dhanpat Rai Publicating

Standard Books:

1. Engineering Chemistry by Jain and Jain; DhanpatRaiPublicating Co.
2. Engineering Chemistry by Shikha Agarwal; Cambridge University Press, 2015 edition.

Reference Books:

1. Engineering Chemistry of Wiley India Pvt. Ltd., Vairam and others, 2014 edition (second).
2. Engineering Chemistry by PrasanthRath, Cengage Learning, 2015 edition.
3. A text book of engineering Chemistry by S. S. Dara; S. Chand & Co Ltd., Latest Edition
4. Applied Chemistry by H.D. Gesser, Springer Publishers
5. Text book of Nano-science and nanotechnology by B.S. Murthy, P. Shankar and others, UniversityPress,IM.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

PROGRAMMING FOR PROBLEM SOLVING USING C

Course Objectives:

- To learn the fundamentals of computers.
- To understand the various steps in program development.
- To learn the syntax and semantics of C programming language.
- To learn the usage of structured programming approach in solving problems.

Course Outcomes: The student will learn

- To write algorithms and to draw flowcharts for solving problems.
- To convert the algorithms/flowcharts to C programs.
- To code and test a given logic in C programming language.
- To decompose a problem into functions and to develop modular reusable code.
- To use arrays, pointers, strings and structures to write C programs.
- Searching and sorting problems.

UNIT - I: Introduction to Programming

Introduction to Algorithms: steps to solve logical and numerical problems. Representation of Algorithm, Flowchart/Pseudo code with examples, Program design and structured programming
Introduction to C Programming Language: variables (with data types and space requirements), Syntax and Logical Errors in compilation, object and executable code , Operators, expressions and precedence, Expression evaluation, Storage classes (auto, extern, static and register), type conversion, The main method and command line arguments

UNIT – II: Bitwise operations

Bitwise operations: Bitwise AND, OR, XOR and NOT operators

Conditional Branching and Loops: Writing and evaluation of conditionals and consequent branching with if, if-else, switch-case, ternary operator, goto, Iteration with for, while, do-while loops

I/O: Simple input and output with scanf and printf, formatted I/O, Introduction to stdin, stdout and stderr. Command line arguments

UNIT - III: Arrays, Strings, Structures and Pointers:

Arrays: one- and two-dimensional arrays, creating, accessing and manipulating elements of arrays

Strings: Introduction to strings, handling strings as array of characters, basic string functions available in C (strlen, strcat, strcpy, strstr etc.), arrays of strings

Structures: Defining structures, initializing structures, unions, Array of structures

Pointers: Idea of pointers, Defining pointers, Pointers to Arrays and Structures, Use of Pointers in self-referential structures, usage of self-referential structures in linked list (no implementation)

Enumeration data type

UNIT - IV: Preprocessor and File handling in C:

Preprocessor: Commonly used Preprocessor commands like include, define, undef, if, ifdef, ifndef

Files: Text and Binary files, Creating and Reading and writing text and binary files, Appending data to existing files, Writing and reading structures using binary files, Random access using fseek, ftell and rewind functions.

UNIT - V: Function and Dynamic Memory Allocation:

Functions: Designing structured programs, Declaring a function, Signature of a function,

Parameters and return type of a function, passing parameters to functions, call by value, Passing arrays to functions, passing pointers to functions, idea of call by reference, Some C standard functions and libraries.

UNIT VI: Strings

Strings – Concepts, C Strings, String Input / Output functions, string manipulation functions, string /data conversion. Input and Output – Concept of a file, streams, text files and binary files, Differences between text and binary files, State of a file, Opening and Closing files, file input / output functions (standard library input / output functions for files), file status functions (error handling), Positioning functions.

TEXT BOOKS:

1. Byron Gottfried, Schaum's Outline of Programming with C, McGraw-Hill
2. B.A. Forouzan and R.F. Gilberg C Programming and Data Structures, Cengage Learning, (3rd Edition)

REFERENCE BOOKS:

1. Brian W. Kernighan and Dennis M. Ritchie, The C Programming Language, Prentice
2. Hall of India
3. R.G. Dromey, How to solve it by Computer, Pearson (16th Impression)
4. Programming in C, Stephen G. Kochan, Fourth Edition, Pearson Education.
5. Herbert Schildt, C: The Complete Reference, Mc Graw Hill, 4th Edition

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

ENGINEERING DRAWING

Year 1 Semester 1

REGULATION : RM:21

L 3 T 0 P 0 C 3

Course Objectives:

- Engineering drawing being the principal method of communication for engineers, the objective is to introduce the students
- To recognize the Bureau of Indian Standards of Engineering Drawing
- The techniques of constructing the various types of polygons, curves and scales.
- The objective is also to visualize and Represent the 3D objects in 2D planes with proper dimensioning, scaling etc.
- To develop an ability to get familiarized with orthographic projections and isometric views of solid objects

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

CO 1: Represent the geometrical objects considering BIS standards.

CO2: Comprehend the basics of orthographic projections

CO3: To deduce orthographic Projections of a point and a line at different Orientations.

CO4: Visualize geometrical planes of different positions in real life environment

CO5: Imagine orthographic views of various solid objects at different orientations

CO6: Recognize the significance of isometric drawing to relate 2D environment with 3D environment

UNIT – I

INTRODUCTION TO ENGINEERING DRAWING:

INTRODUCTION: Principles of Engineering Graphics and their significance - Drawing Instruments and their use-Conventions in Drawing- Lettering and Dimensioning – BIS conventions

Geometrical Constructions: Constructing regular polygons by general methods, inscribing and describing polygons on circles.

Engineering Curves: Conic Sections- Ellipse, Parabola, Hyperbola and rectangular hyperbola- General method and other methods; Cycloid, Epi-Cycloid and Hypo-Cycloid; Involute

UNIT – II:

Scales: Plain scales, diagonal scales and vernier scales

Principle of orthographic projection-Method of Projections – First and third angle projection methods-Projections of Points

UNIT - III:

ORTHOGRAPHIC PROJECTIONS:

Projections of straight lines inclined to both the planes, determination of true lengths, angle Of inclination and traces.

UNIT - IV:

PROJECTIONS OF PLANES:

Regular planes perpendicular/parallel to one reference plane and inclined to

the other reference plane; inclined to both the reference planes.

UNIT - V:

PROJECTIONS OF SOLIDS:

Introduction, Regular Polyhedral, Solids of Revolution, Projection of solids in simple position - Axis inclined to one of the reference planes and parallel to the other.

UNIT VI:

ISOMETRIC VIEWS:

Introduction-theory of isometric projection, isometric views, isometric axes, scale, lines and planes-Isometric view of prism, pyramid, cylinder and cone-non isometric lines-methods to generate an isometric drawing

TRANSFORMATION OF PROJECTIONS:

Conversion of Orthographic Projections to Isometric Views of composite objects, Conversion of Isometric Views to Orthographic Projections.

TEXT BOOKS:

1.N. D. Bhatt, Engineering Drawing, 51th Revised and Enlarged Edition, Charotar publishers, 2012.

REFERENCE BOOKS:

- 1.Narayana K L, Kanniah P, Textbook on Engineering Drawing, 2nd Edition, SciTech publishers.
- 2.R.K.Dhawan, Engineering Drawing, S.Chand Company LTD
- 3.Venugopal, Engineering Drawing and Graphics, New Age publishers
- 4.Dhananjay A. Jolhe, Engineering Drawing, Tata McGraw Hill Publishers
- 5.N.S.Parthasarathy, Vela Murali, Engineering Drawing, Oxford Higher Education

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

ENGLISH COMMUNICATION SKILLS LAB

Year 1 Semester 1

REGULATION : RM:21

L 0 T 0 P 3 C 2

Course Objectives:-

- To improve the communicative ability.

- To enhance the general conversational skills in different socio-cultural contexts.
- To strengthen their professional skills.
- To instill confidence and make them competent enough to express themselves fluently.
- To expose the students to various spoken skills.
- To understand the role of communication in personal & professional success.

Course outcomes:-

- Better pronunciation and accent
- Ability to use functional English
- Competency in analytical skills and problem solving skills
- Ability to develop effective communication skills (spoken and written).
- Ability to develop effective presentation skills.
- Ability to develop speaking skills.

UNIT 1:

Phonetics

Introduction to sounds of English. Phonetic transcription of simple words.

UNIT II:

JAM (just a minute)

Practice personal and peer profiles

UNIT III:

Oral presentation skills

Public speaking-practice

UNIT IV:

Group Discussions.

GD practice & modalities

UNIT V:

Interview skills

Mock interview skills & practice

UNIT VI: Dialogues.

Conversation practice

References:

1. Exercises in Spoken English Part 1,2,3,4, OUP and CIEFL.
2. English Pronunciation in use- Mark Hancock, Cambridge University Press.
3. English Phonetics and Phonology-Peter Roach, Cambridge University Press.
4. English Pronunciation in use- Mark Hewings, Cambridge University Press.
5. English Pronunciation Dictionary- Daniel Jones, Cambridge University Press.
6. English Phonetics for Indian Students- P. Bala Subramanian, Mac Millan

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

APPLIED CHEMISTRY LAB

Year 1 Semester 1

REGULATION : RM:21

L 0 T 0 P 3 C 2

Course objectives:

*The experiments introduce volumetric analysis ,Redox titrations and EDTA Titrations.

*A Few instrumental methods of chemical analysis.

Course outcomes:

At the end of the course ,student will be able to

Co:The student is exposed to different methods of chemical analysis and use of some commonly employed instruments. They thus acquire some experimental skills.

SYLLABUS

Introduction to Chemistry laboratory – Molarity, normality, primary, secondary standard solutions, volumetric titrations, quantitative analysis

1. Determination of HCl using standard Na_2CO_3 solution.
2. Determination of alkalinity of a sample containing Na_2CO_3 and NaOH.
3. Determination of Mn^{+2} using standard oxalic acid solution.
4. Determination of ferrous iron using standard $\text{K}_2\text{Cr}_2\text{O}_7$ solution.
5. Determination of Cu^{+2} using standard hypo solution.
6. Determination of temporary and permanent hardness of water using standard EDTA solution.
7. Determination of Fe^{+3} by a colorimetric method.
8. Determination of the concentration of acetic acid using sodium hydroxide (pH-metrymethod)
9. Determination of iso-electric point of amino acids using pH-metry method/conduct metric method.
10. Determination of the concentration of strong acid vs strong base (by conduct metric method).
11. Determination of strong acid vs strong base (by potentiometric method).
12. Determination of PH of given Sample.
13. Determination of CaCO_3 present in an egg shell.
14. Estimation of Vitamin C.
15. Preparation of nylon-6, 6 and Bakelite (demonstration only).

Of the above experiments at-least 10 assessment experiments should be completed

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

C PROGRAMMING LAB

Year 1 Semester 1

REGULATION : RM:21

L 0 T 0 P 3 C 2

Course Objectives:

- Apply the principles of C language in problem solving.
- To design flowcharts, algorithms and knowing how to debug programs.
- To design & develop of C programs using arrays, strings pointers & functions.
- To review the file operations, preprocessor commands.

Course Outcomes:

By the end of the Lab, the student

- Gains Knowledge on various concepts of a C language.
- Able to draw flowcharts and write algorithms.
- Able to design and development of C problem solving skills.
- Able to design and develop modular programming skills.
- Able to trace and debug a program
- Able to implement file operations

Exercise 1:

1. Write a C program to print a block F using hash (*), where the F has a height of six characters and width of five and four characters.
2. Write a C program to compute the perimeter and area of a rectangle with a height of 10 inches and width of 6 inches.
3. Write a C program to print even numbers up to 500.

Exercise 2:

1. Write a C program to calculate the distance between the two points.
2. Write a C program that accepts 4 integers p, q, r, s from the user where r and s are positive and p is even. If q is greater than r and s is greater than p and if the sum of r and s is greater than the sum of p and q print "Correct values", otherwise print "Wrong values".

Exercise 3:

1. Write a C program to convert a string to a long integer.
2. Write a program in C which is a Menu-Driven Program to compute the area of the various geometrical shape.
3. Write a C program to calculate the factorial of a given number.

Exercise 4:

1. Write a program in C to display the n terms of even natural number and their sum.
2. Write a program in C to display the n terms of harmonic series and their sum. $1 + 1/2 + 1/3 + 1/4 + 1/5 \dots 1/n$ terms.
3. Write a C program to check whether a given number is an Armstrong number or not.

Exercise 5:

1. Write a program in C to print all unique elements in an array.
2. Write a program in C to separate odd and even integers in separate arrays.
3. Write a program in C to sort elements of array in ascending order.

Exercise 6:

1. Write a program in C for multiplication of two square Matrices.
2. Write a program in C to find transpose of a given matrix.

Exercise 7:

1. Write a program in C to search an element in a row wise and column wise sorted matrix.
2. Write a program in C to print individual characters of string in reverse order.

Exercise 8:

1. Write a program in C to compare two strings without using string library functions.
2. Write a program in C to copy one string to another string.

Exercise 9:

1. Write a C Program to Store Information Using Structures with Dynamically Memory Allocation
2. Write a program in C to demonstrate how to handle the pointers in the program.

Exercise 10:

1. Write a program in C to demonstrate the use of & (address of) and *(value at address) operator.
2. Write a program in C to add two numbers using pointers.

Exercise 11:

1. Write a program in C to add numbers using call by reference.
2. Write a program in C to find the largest element using Dynamic Memory Allocation.

Exercise 12:

1. Write a program in C to swap elements using call by reference.
2. Write a program in C to count the number of vowels and consonants in a string using a pointer.

Exercise 13:

1. Write a program in C to show how a function returning pointer.
2. Write a C program to find sum of n elements entered by user. To perform this program, allocate memory dynamically using malloc () function.

Exercise 14:

1. Write a C program to find sum of n elements entered by user. To perform this program, allocate memory dynamically using calloc () function. Understand the difference between the above two programs
2. Write a program in C to convert decimal number to binary number using the function.

Exercise 15:

1. Write a program in C to check whether a number is a prime number or not using the function.
2. Write a program in C to generate 20 random numbers using arrays between 100 to 200.

Exercise 16:

1. Write a program in C to append multiple lines at the end of a text file.
2. Write a program in C to copy a file in another name.
3. Write a program in C to concatenate two files.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

APPLIED PHYSICS

Year 1 Semester 11

REGULATION : RM:21

L 3 T 0 P 0 C 3

Course Objectives: Objective of the course is to impart

- Knowledge of fundamentals of Physics which helps them in the study of advanced topics of Engineering.
- Develop analytical capability and understand various Engineering concepts.

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

- Gain knowledge of applications of lasers and optical fibers in various fields.
- Classify crystal systems and analyze the crystalline structure using various X-ray diffraction techniques.
- Classify magnetic and dielectric materials and their Engineering applications.
- The students will be able to analyse the physical significance of wave function and apply Schrödinger's wave equation for energy values of a free particle.
- Impart knowledge of architectural acoustics and Study of Ultrasonic.
- Obtain the concept of holes and electrons in semiconductors.

Unit-I: Lasers and Fiber optics

Lasers: Introduction – Characteristics of laser – Spontaneous and Stimulated emissions of radiation — Population inversion –Lasing action- Pumping mechanisms – Ruby laser – He-Ne laser - Applications of lasers.

Fiber optics: Introduction –Principle of optical fiber- Acceptance Angle-Numerical Aperture- Classification of optical fibers based on refractive index profile and modes –Applications.

Unit-II: Crystallography and X-ray diffraction

Crystallography: Space lattice, Basis, Unit Cell and lattice parameters – Bravais Lattice – crystal systems (3D) – coordination number - packing fraction of SC, BCC & FCC structures.

X- ray diffraction: Crystal planes and directions - Miller indices–Separation between successive (h k l) planes–Bragg's law.

Unit- III: Engineering Materials

Unit-III-Magnetism:Introduction-Magnetization, Magnetic susceptibility and permeability -Origin of Magnetic Moment-Classification of Magnetic Materials (Dia, Para and Ferro) -Domain Theory – hysteresis-soft and hard magnetic materials.

DIELECTRICS: Introduction - electronic, ionic polarizations and Orientation Polarization (Qualitative). Inter fields-Classius-Mosotti equation.

Unit-IV- Quantum Mechanics: Introduction – Matter waves – de Broglie's hypothesis– Heisenberg's Uncertainty Principle –Interpretation of wave function – Schrödinger Time Independent and Time Dependent wave equations – Particle in a potential box. **(09hrs)**

Unit-V: Acoustics and Ultrasonics

Acoustics: Introduction – requirements of acoustically good hall– Reverberation – Reverberation time– Sabine's formula (Derivation using growth and decay method).

Ultrasonics: Introduction - Properties - Production by magnetostriction and piezoelectric methods – Non Destructive Testing - Applications.

Unit – VI: Semiconductors

Semiconductors: Introduction- Intrinsic semiconductors – Density of charge carriers – Fermi level – extrinsic semiconductors – density of charge carriers –Drift and diffusion currents -Hall effect – Hall coefficient –Applications of Hall effect. Applications of Semiconductors.

Text books:

1. Engineering Physics – Dr. M.N. Avadhanulu & Dr. P.G. Kshirsagar, S. Chand and Company
2. Engineering physics – D.K. Battacharya and Poonam Tandon, Oxford University press.
3. Engineering Physics by P.K.Palanisamy SciTech publications.

Reference Books:

1. Fundamentals of Physics – Halliday, Resnick and Walker, John Wiley & Sons
2. Engineering Physics – M.R.Srinivasan, New Age Publications
3. Engineering Physics – D K Pandey, S. Chaturvedi, Cengage Learning
4. Engineering Physics - Sanjay D. Jain, D. Sahasrambudhe and Girish, University Press

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

NUMERICAL METHODS & VECTOR CALCULUS

Course objectives:

- To understand the basic numerical methods to solve non linear algebraic equations.
- To demonstrate the use of different numerical techniques for carrying out numerical integration
- To Illustrate the types of Integration over the lines, surface and volumes

Course Outcomes : At the end of the course, the student will learn

CO1: Evaluate the approximate roots of polynomial and transcendental equations by different algorithms

CO2: Apply Newton's forward & backward interpolation and Lagrange's formulae for equal and unequal intervals

CO3: Discuss numerical integral techniques to different Engineering problems. Apply different algorithms for approximating the solutions of ordinary differential equations with initial conditions to its analytical computations

CO4: Calculate double integration techniques in evaluating areas bounded by region

CO5: Interpret the physical meaning of different operators such as gradient, curl and divergence

CO6: Estimate the work done against a field, circulation and flux using vector calculus

UNIT I: Iterative methods

Introduction , Bisection method , Method of false position ,Iteration method , Newton-Raphson method ,Jacobi and Gauss-Seidel methods for solving system of equations numerically.

UNIT II: Interpolation

Introduction , Errors in polynomial interpolation , Finite differences , Forward differences , Backward differences , Central differences , Relations between operators , Newton's forward and backward formulae for interpolation , Interpolation with unequal intervals , Lagrange's interpolation formula , Newton's divide difference formula.

UNIT III: Numerical integration, Solution of ordinary differential equations with initial conditions

Numerical integration, Trapezoidal rule , Simpson's 1/3rd and 3/8th rule, Solution of initial value problems by Taylor's series ,Picard's method of successive approximations , Euler's method ,Runge-Kutta method (fourth order).

UNIT IV: Multiple integrals

Double and Triple integrals , Change of order of integration in double integrals , Change of variables to polar coordinates.

Applications: Finding Areas and Volumes.

UNIT V: Vector Differentiation

Gradient , Directional derivative , Divergence , Curl , Scalar Potential, Vector identities , Laplacian and second order operators.

UNIT VI: Vector Integration

Line integral , Work done , Area , Surface and volume integrals .

Vector integral theorems: Greens, Stokes and Gauss Divergence theorems (without proof).

Text Books:

2. **B. S. Grewal**, Higher Engineering Mathematics, 43rd Edition, Khanna Publishers.
2. **B. V. Ramana**, Higher Engineering Mathematics, 2007 Edition, Tata Mc. Graw Hill Education.

Reference Books:

3. **Erwin Kreyszig**, Advanced Engineering Mathematics, 10th Edition, Wiley-India
4. **N.P.BALI &Dr.MANISH GOYAL**, A Text book of Engineering Mathematics, Lakshmi Publications, 9th Edition, 2014.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

DATA STRUCTURES

Year 1 Semester 11

REGULATION : RM:21

L 3 T 0 P 0 C 3

Course Objectives:

- Exploring basic data structures such as stacks and queues.
- Introduces a variety of data structures such as hash tables, search trees, tries, heaps, graphs.
- Introduces sorting and pattern matching algorithms

Course Outcomes:

- Ability to select the data structures that efficiently model the information in a problem.
- Ability to assess efficiency trade-offs among different data structure implementations or combinations.
- Implement and know the application of algorithms for sorting and pattern matching.
- Design programs using a variety of data structures, including hash tables, binary and general tree structures, search trees, tries, heaps, graphs, and AVL-trees.
- To develop application using data structures.
- To improve the logical ability

UNIT-I

Time and space complexity, Data Structures – Introduction to Data Structures, abstract data types, Linear list – singly linked list implementation, insertion, deletion and searching operations on linear list, circular linked list implementation, Double linked list implementation, insertion, deletion and searching operations. Applications of linked lists.

UNIT –II

Stacks-Operations, array and linked representations of stacks, stack applications -infix to postfix conversion, postfix expression evaluation, recursion implementation.

UNIT-III

Queues-operations, array and linked representations. Circular Queue operations, Dequeues, applications of queues.

UNIT-IV

Searching and Sorting – Sorting- selection sort, bubble sort, insertion sort, quick sort, merge sort, shell sort, radix sort, Searching-linear and binary search methods, comparison of sorting and searching methods.

UNIT-V

Trees – Definitions, tree representation, properties of trees, Binary tree, Binary tree representation, binary tree properties, binary tree traversals, binary tree implementation, applications of trees.

UNIT – VI

Graphs: Graph Implementation Methods. Graph Traversal Methods.

Pattern Matching and Tries: Pattern matching algorithms-Brute force, the Boyer –Moore algorithm, the Knuth-Morris-Pratt algorithm, Standard Tries, Compressed Tries, Suffix tries.

TEXT BOOKS:

1. Fundamentals of Data structures in C, 2nd Edition, E.Horowitz, S.Sahni and Susan Anderson-Freed, Universities Press.
2. Data structures A Programming Approach with C, D.S.Kushwaha and A.K.Misra, PHI.

REFERENCE BOOKS:

1. Data structures: A Pseudocode Approach with C, 2nd edition,
2. R.F.GilbergAndB.A.Forouzan, CengageLearning.
3. Data structures and Algorithm Analysis in C, 2nd edition, M.A.Weiss, Pearson.
4. Data Structures using C, A.M.Tanenbaum, Y. Langsam, M.J.Augenstein, Pearson.
5. Data structures and Program Design in C, 2nd edition, R.Kruse, C.L.Tondo and B.Leung, Pearson

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

COMPUTER ORGANIZATION & ARCHITECTURE

Course Objectives:

- The purpose of the course is to introduce principles of computer organization and the basic architectural concepts.
- It begins with basic organization, design, and programming of a simple digital computer and introduces simple register transfer language to specify various computer operations.
- Topics include computer arithmetic, instruction set design, micro programmed control unit, pipelining and vector processing, memory organization and I/O systems, and multiprocessors

Course Outcomes:

- Understand the basics of instructions sets and their impact on processor design.
- Demonstrate an understanding of the design of the functional units of a digital computer system.
- Evaluate cost performance and design trade-offs in designing and constructing a computer processor including memory.
- Design a pipeline for consistent execution of instructions with minimum hazards.
- Recognize and manipulate representations of numbers stored in digital computers
- Expose different ways of communicating with I/O devices and standard I/O interfaces.

UNIT – I

Digital Computers: Introduction, Block diagram of Digital Computer, Definition of Computer Organization, Computer Design and Computer Architecture.

Register Transfer Language and Micro operations: Register Transfer language, Register Transfer, Bus and memory transfers, Arithmetic Micro operations, logic micro operations, shift micro operations, Arithmetic logic shift unit.

UNIT-II

Basic Computer Organization and Design: Instruction codes, Computer Registers Computer instructions, Timing and Control, Instruction cycle, Memory Reference Instructions, Input – Output and Interrupt.

UNIT – III

Micro programmed Control: Control memory, Address sequencing, micro program example, design of control unit.

Central Processing Unit: General Register Organization, Instruction Formats, Addressing modes, Data Transfer and Manipulation, Program Control.

UNIT – IV

Data Representation: Data types, Complements, Fixed Point Representation, Floating Point Representation.

Computer Arithmetic: Addition and subtraction, multiplication Algorithms, Division Algorithms, Floating – point Arithmetic operations. Decimal Arithmetic unit, Decimal Arithmetic operations.

UNIT – V

Input-Output Organization: Input-Output Interface, Asynchronous data transfer, Modes of Transfer, Priority Interrupt Direct memory Access.

Memory Organization: Memory Hierarchy, Main Memory, Auxiliary memory, Associate Memory, Cache Memory.

UNIT – VI

Reduced Instruction Set Computer: CISC Characteristics, RISC Characteristics.

Pipeline and Vector Processing: Parallel Processing, Pipelining, Arithmetic Pipeline, Instruction Pipeline, RISC Pipeline, Vector Processing, Array Processor.

Multi Processors: Characteristics of Multiprocessors, Interconnection Structures, Interprocessor arbitration, Interprocessor communication and synchronization, Cache Coherence.

TEXT BOOK:

1. Computer System Architecture – M. Moris Mano, Third Edition, Pearson/PHI.

REFERENCE BOOKS:

1. Computer Organization – Car Hamacher, Zvonks Vranesic, Safea Zaky, Vth Edition, McGrawHill.
2. Computer Organization and Architecture – William Stallings Sixth Edition, Pearson/PHI.
3. Structured Computer Organization – Andrew S. Tanenbaum, 4th Edition, PHI/Pearson.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

DIGITAL LOGIC DESIGN

Year 1 Semester 11

REGULATION : RM:21

L 3 T 0 P 0 C 3

UNIT-I: Number Systems

Binary, Octal, Decimal, Hexadecimal Number Systems. Conversion of Numbers From One Radix To Another Radix, r 's Complement and $(r-1)$'s Complement Subtraction of Unsigned Numbers, Problems, Signed Binary Numbers, Weighted and Non weighted codes

UNIT II: Logic Gates And Boolean Algebra

Basic Gates NOT, AND, OR, Boolean Theorems, Complement And Dual of Logical Expressions, Universal Gates, Ex-Or and Ex-Nor Gates, SOP, POS, Minimizations of Logic Functions Using Boolean Theorems, Two level Realization of Logic Functions Using Universal Gates

Gate Level Minimization: Karnaugh Map Method (K-Map): Minimization of Boolean Functions maximum upto Four Variables, POS and SOP, Simplifications With Don't Care Conditions Using K-Map.

UNIT III: Combinational Logic Circuits

Design of Half Adder, Full Adder, Half Subtractor, Full Subtractor, Ripple Adders and Subtractors, Ripple Adder/Subtractor Using Ones and Twos Complement Method. Design of Decoders, Encoders, Multiplexers, Demultiplexers, Higher Order Demultiplexers and Multiplexers, Priority Encoder, Code Converters, Magnitude Comparator.

UNIT IV: Introduction to Sequential Logic Circuits

Classification of Sequential Circuits, Basic Sequential Logic Circuits: Latch and Flip-Flop, RS- Latch Using NAND and NOR Gates, Truth Tables. RS, JK, T and D Flip Flops, Truth and Excitation Tables, Conversion of Flip Flops. Flip Flops With Asynchronous Inputs (Preset and Clear).

UNIT V: Registers and Counters

Design of Registers, Buffer Register, Control Buffer Registers, Bidirectional Shift Registers, Universal Shift Register, Design of Ripple Counters, Synchronous Counters and Variable Modulus Counters, Ring Counter, Johnson Counter.

UNIT VI: Introduction to Programmable Logic Devices (PLOs)

PLA, PAL, PROM. Realization of Switching Functions Using PROM, PAL and PLA. Comparison of PLA, PAL and PROM.

TEXT BOOKS :

1. Digital Design ,4/e, M.Morris Mano, Michael D Ciletti, PEA
2. Fundamentals of Logic Design, 5/e, Roth, Cengage

REFERENCE BOOKS

1. Switching and Finite Automata Theory,3/e,Kohavi, Jha, Cambridge.
2. Digital Logic Design, Leach,

Malvino, Saha, TMH 3. Modern Digital
Electronics, R.P. Jain, TMH

Course Objective: Objective of the course is to impart

- Training Engineering graduates to handle instruments and their usage methods to improve the accuracy of measurements.

CO1: Outcomes: The student is exposed to different methods of chemical analysis and use of some commonly employed instruments. They thus acquire some experimental skills.

CO2: Implement the basic principles of Mechanics to measure different physical parameters.

CO3: Enhance the knowledge of Usage of electronic devices in various applications.

SYLLABUS

1. Laser: Determination of wavelength using diffraction grating.
2. Study of variation of magnetic field along the axis of a current carrying circular coil by Stewart & Gee's method
3. Determination of velocity of sound – volume resonator.
4. Determination of ultrasonic velocity in given liquid (Acoustic grating).
5. Determination of dielectric constant using charging and discharging method.
6. Determination of Acceleration due to Gravity and Radius of Gyration- Compound Pendulum.
7. Determine the Velocity of sound in air by using volume Resonator.
8. Rigidity modulus of material of a wire-dynamic method (Torsional pendulum).
9. Determination of numerical aperture and acceptance angle of an optical fiber.
10. Determination of thickness of thin object by wedge method.
11. Determination of radius of curvature of given plano convex lens by Newton's rings.
12. Determination of wavelengths of different spectral lines in mercury spectrum using diffraction grating in normal incidence configuration.
13. Determination of dispersive power of the prism.
14. Sonometer: Verification of laws of string.
15. Meled's experiment-Transverse and Longitudinal modes.

References:

1. S. Balasubramanian, M.N. Srinivasan "A Text book of Practical Physics"- S Chand Publishers, 2017.

DATA STRUCTURES LAB

Year 1 Semester 11

REGULATION : RM:21

L 0 T 0 P 3 C 2

Course Objectives:

- It covers various concepts of C programming language
- It introduces searching and sorting algorithms
- It provides an understanding of data structures such as stacks and queues.

Course Outcomes:

- Ability to develop C programs for computing and real-life applications using basic elements like control statements, arrays, functions, pointers and strings, and data structures like stacks, queues and linked lists.
- Ability to Implement searching and sorting algorithms

LIST OF EXPERIMENTS

1. Write a program that uses functions to perform the following operations on singly linked list:
i) Creation ii) Insertion iii) Deletion iv) Traversal
2. Write a program that uses functions to perform the following operations on doubly linked list:
i) Creation ii) Insertion iii) Deletion iv) Traversal
3. Write a program that uses functions to perform the following operations on circular linked list:
i) Creation ii) Insertion iii) Deletion iv) Traversal
4. Write a program that implement stack (its operations) using
i) Arrays ii) Pointers
5. Write a program that implement Queue (its operations) using
i) Arrays ii) Pointers
6. Write a program that implements the following sorting methods to sort a given list of integers in ascending order
i) Bubble sort ii) Selection sort iii) Insertion sort
7. Write a program that use both recursive and non recursive functions to perform the following searching operations for a Key value in a given list of integers:
i) Linear search ii) Binary search
8. Write a program to implement the tree traversal methods.
9. Write a program to implement the graph traversal methods.

TEXTBOOKS:

1. Fundamentals of Data Structures in C, 2nd Edition, E. Horowitz, S. Sahni and Susan Anderson Freed, *Universities Press*.

2. Data Structures using C – A. S. Tanenbaum, Y. Langsam, and M. J. Augenstein, *PHI/Pearson Education*.

REFERENCE:

1. Data Structures: A Pseudocode Approach with C, 2nd Edition, R. F. Gilberg and B. A. Forouzan, Cengage *Learning*.

Course Objectives:

- Identify the students should work on working PC to disassemble and assemble to working condition and install Windows and Linux on the same PC.
- Students are suggested to work similar tasks in the Laptop scenario wherever possible
- Usage of web browsers, email, newsgroups and discussion forums would be covered.
- Awareness of cyber hygiene, i.e., protecting the personal computer from getting infected with the viruses, worms and other cyber attacks would be introduced.
- Enable the students in crafting professional word documents, excel spread sheets.
- Designing power point presentations and personal web sites using the Microsoft suite of office tools and Latex.

PC Hardware

Exercise 1: Identify the peripherals of a computer, components in a CPU and its functions. Draw the blockdiagram of the CPU along with the configuration of each peripheral and submit to your instructor.

Exercise 2: Every student should disassemble and assemble the PC back to working condition. Lab instructors should verify the work and follow it up with a Viva. Also students need to go through the video which shows the process of assembling a PC. A video would be given as part of the course content.

Exercise 3: Every student should individually install MS windows on the personal computer. Lab instructor should verify the installation and follow it up with a Viva.

Exercise4: Every student should install Linux on the computer. This computer should have windows installed. The system should be configured as dual boot with both windows and Linux. Lab instructors should verify the installation and follow it up with a Viva

Exercise 5: Hardware Troubleshooting: Students have to be given a PC which does not boot due to improper assembly or defective peripherals. They should identify the problem and fix it to get the computer back to working condition. The work done should be verified by the instructor and followed up with a Viva.

Exercise 6: Software Troubleshooting: Students have to be given a malfunctioning CPU due to system software problems. They should identify the problem and fix it to get the computer back to working condition. The work done should be verified by the instructor and followed up with a Viva.

Internet & World Wide Web

Exercise 7: Orientation & Connectivity Boot Camp: Students should get connected to their Local Area Network and access the Internet. In the process they configure the TCP/IP setting. Finally students should demonstrate, to the instructor, how to access the websites and email. If there is no internet connectivity preparations need to be made by the instructors to simulate the WWW on the LAN.

Exercise 8: Web Browsers, Surfing the Web: Students customize their web browsers with the LAN proxy settings, bookmarks, search toolbars and pop up blockers. Also, plug-ins like Macromedia Flash and JRE for applets should be configured.

Exercise 9: Search Engines & Netiquette: Students should know what search engines are and how to use the search engines. A few topics would be given to the students for which they need to search on Google. This should be demonstrated to the instructors by the student.

Exercise 10: Cyber Hygiene: Students would be exposed to the various threats on the internet and would be asked to configure their computer to be safe on the internet. They need to first install antivirus software, configure their personal firewall and windows update on their computer. Then they need to customize their browsers to block pop ups, block active x downloads to avoid viruses and/or worms.

LaTeX and WORD

Exercise 11 – Word Orientation: The mentor needs to give an overview of LaTeX and Microsoft (MS) office 2007/ equivalent (FOSS) tool word: Importance of LaTeX and MS office 2007/ equivalent (FOSS) tool Word as word Processors, Details of the four tasks and features that would be covered in each, Using LaTeX and word – Accessing, overview of toolbars, saving files, Using help and resources, rulers, format painter in word.

Exercise 12: Using LaTeX and Word to create project certificate. Features to be covered:- Formatting Fonts in word, Drop Cap in word, Applying Text effects, Using Character Spacing, Borders and Colors, Inserting Header and Footer, Using Date and Time option in both LaTeX and Word.

Exercise 13: Creating project abstract Features to be covered:-Formatting Styles, Inserting table, Bullets and Numbering, Changing Text Direction, Cell alignment, Footnote, Hyperlink, Symbols, Spell Check, Track Changes.

Exercise 14 : Creating a Newsletter : Features to be covered:- Table of Content, Newspaper columns, Images from files and clipart, Drawing toolbar and Word Art, Formatting Images, Textboxes, Paragraphs and Mail Merge in word.

Excel

Excel Orientation: The mentor needs to tell the importance of MS office 2007/ equivalent (FOSS) tool Excel as a Spreadsheet tool, give the details of the four tasks and features that would be covered in each. Using Excel – Accessing, overview of toolbars, saving excel files, Using help and resources.

Exercise 15: Creating a Scheduler - Features to be covered: Gridlines, Format Cells, Summation, auto fill, Formatting Text

Exercise 16 : Calculating GPA - .Features to be covered:- Cell Referencing, Formulae in excel – average, std.deviation, Charts, Renaming and Inserting worksheets, Hyper linking, Count function, LOOKUP/VLOOKUP

Exercise 17: Performance Analysis - Features to be covered:- Split cells, freeze panes, group and outline, Sorting, Boolean and logical operators, Conditional formatting

LaTeX and MS/equivalent (FOSS) tool Power Point

Exercise 1: Students will be working on basic power point utilities and tools which help them create basic power point presentation. Topic covered during this week includes: - PPT Orientation, Slide Layouts, Inserting Text, Word Art, Formatting Text, Bullets and Numbering, Auto Shapes, Lines and Arrows in both LaTeX and PowerPoint.

Exercise 18: Second week helps students in making their presentations interactive. Topic covered during this week includes: Hyperlinks, Inserting –Images, Clip Art, Audio, Video, Objects, Tables and Charts.

Exercise 19: Concentrating on the in and out of Microsoft power point and presentations in LaTeX. Helps them learn best practices in designing and preparing power point presentation. Topic covered during this week includes: - Master Layouts (slide, template, and notes), Types of views (basic, presentation, slide slotter, notes etc), and Inserting – Background, textures, Design Templates, Hidden slides.

REFERENCE BOOKS:

1. Comdex Information Technology course tool kit Vikas Gupta, *WILEY Dreamtech*
2. The Complete Computer upgrade and repair book, 3rd edition Cheryl A Schmidt, *WILEY Dreamtech*
3. Introduction to Information Technology, ITL Education Solutions limited, *Pearson Education*.
4. PC Hardware - A Handbook – Kate J. Chase *PHI* (Microsoft)
5. LaTeX Companion – Leslie Lamport, *PHI/Pearson*.
6. IT Essentials PC Hardware and Software Companion Guide Third Edition by David Anfinson and Ken Quamme. – *CISCO Press, Pearson Education*.
7. IT Essentials PC Hardware and Software Labs and Study Guide Third Edition by Patrick Regan – *CISCO Press, Pearson Education*.

II Year I Semester		L	T	P	C
		3	0	0	3
TRANSFORM TECHNIQUES (Common to All Branches)					

Course objectives:

- To familiarize the techniques in partial differential equations
- To furnish the learners with basic concepts and techniques sat plus two level to lead them into advanced level by handling various real-world application

COURSE OUTCOMES: At the end of the course, the student will learn

- Apply the Laplace transform for solving differential equations.
- Find or compute the Fourier series of periodic signals.
- Know and be able to apply integral expressions for the forwards and inverse Fourier transform to arrange of non-periodic wave forms.
- Identify solution methods for partial differential equations that model physical processes.
- To know the solution of second and higher order partial differential equations.

UNIT I: Laplace Transforms:

Laplace transforms of standard functions – Properties -Shifting theorems – Transforms of derivatives and integrals – Inverse Laplace transforms– Convolution theorem (without proof).

Applications: Solving ordinary differential equations (initial value problems) using Laplace transforms

UNIT II : Fourier series:

Introduction – Periodic functions – Fourier series of periodic function –Dirichlet’s conditions – Even and odd functions – Change of interval – Half-range sine and cosine series.

UNIT III: Fourier Transforms:

Fourier integral theorem (without proof) –Fourier sine and cosine integrals –Sine and cosine transforms –inverse transforms –Finite Fourier transforms

UNIT IV: PDE of first order:

Formation of partial differential equations by elimination of arbitrary constants and arbitrary functions – Solutions of first order linear (Lagrange) equation and nonlinear (standard types) equations.

UNIT V: Second order PDE and Applications:

Second order PDE: Solutions of linear partial differential equations with constant coefficient –RHS term of the type e^{ax+by} , $\sin(ax + by)$, $\cos(ax + by)$, $x^m y^n$, Method of separation of variables.

Application: One dimensional Wave Equation.

TEXT BOOKS:

1. B.S.Grewal, Higher Engineering Mathematics, 43rd Edition, Khanna Publishers.
2. B.V.Ramana, Higher Engineering Mathematics, 2007 Edition, Tata Mc.Graw Hill Education

REFERENCE BOOKS:

1. Erwin Kreyszig, Advanced Engineering Mathematics, 10th Edition, Wiley-India.
2. Peter O'Neil Advanced Engineering Mathematics, Cengage.

II Year – I Semester	L	T	P	C
	3	0	0	3
OBJECT ORIENTED PROGRAMMING THROUGH C++				

Course Objectives:

- Describe the procedural and object oriented paradigm with concepts of streams, classes, functions, data and objects
- Understand dynamic memory management techniques using pointers, constructors, destructors
- Describe the concept of function overloading, operator overloading, virtual functions and polymorphism
- Classify inheritance with the understanding of early and late binding, usage of exception handling, generic programming
- Demonstrate the use of various OOPs concepts with the help of programs

Course Outcomes:

By the end of the course, the student

- Classify object oriented programming and procedural programming
- Apply C++ features such as composition of objects, operator overloads, dynamic memory allocation, inheritance and polymorphism, file I/O, exception handling
- Build C++ classes using appropriate encapsulation and design principles
- Apply object oriented or non-object oriented techniques to solve bigger computing problems

UNIT I

Introduction to C++: Difference between C and C++, Evolution of C++, The Object Oriented Technology, Disadvantage of Conventional Programming, Key Concepts of Object Oriented Programming, Advantage of OOP, Object Oriented Language.

UNIT II

Classes and Objects & Constructors and Destructor: Classes in C++, Declaring Objects, Access Specifiers and their Scope, Defining Member Function, Overloading Member Function, Nested class, Constructors and Destructors, Introduction, Constructors and Destructor, Characteristics of Constructor and Destructor, Application with Constructor, Constructor with Arguments parameterized Constructor, Destructors, Anonymous Objects.

UNIT III

Operator Overloading and Type Conversion & Inheritance: The Keyword Operator, Overloading Unary Operator, Operator Return Type, Overloading Assignment Operator (=), Rules for Overloading Operators, Inheritance, Reusability, Types of Inheritance, Virtual Base Classes- Object as a Class Member, Abstract Classes, Advantages of Inheritance, Disadvantages of Inheritance.

UNIT IV

Pointers & Binding Polymorphisms and Virtual Functions: Pointer, Features of Pointers, Pointer Declaration, Pointer to Class, Pointer Object, The this Pointer, Pointer to Derived Classes and Base Class, Binding Polymorphisms and Virtual Functions, Introduction, Binding in C++, Virtual Functions, Rules for Virtual Function, Virtual Destructor.

UNIT V

Generic Programming with Templates & Exception Handling: Definition of class Templates, Normal Function Templates, Over Loading of Template Function, Bubble Sort Using Function Templates, Difference between Templates and Macros, Linked Lists with Templates, Exception Handling, Principles of Exception Handling, The Keywords try throw and catch, Multiple Catch Statements, Specifying Exceptions.

Overview of Standard Template Library, STL Programming Model, Containers, Sequence Containers, Associative Containers, Algorithms, Iterators, Vectors, Lists, Maps.

Text Books:

- 1) A First Book of C++, Gary Bronson, Cengage Learning.
- 2) The Complete Reference C++, Herbert Schildt, TMH.

Reference Books:

- 1) Object Oriented Programming C++, Joyce Farrell, Cengage.
- 2) C++ Programming: from problem analysis to program design, DS Malik, Cengage Learning
- 3) Programming in C++, Ashok N Kamthane, Pearson 2nd Edition

e- Resources:

- 1) <https://nptel.ac.in/courses/106/105/106105151/>
- 2) <https://github.com/topics/object-oriented-programming>

II Year – I Semester	L	T	P	C
	3	0	0	3
OPERATING SYSTEMS				

Course Objectives:

The objectives of this course is to

- Introduce to the internal operation of modern operating systems
- Define, explain, processes and threads, mutual exclusion, CPU scheduling, deadlock, memory management, and file systems
- Understand File Systems in Operating System like UNIX/Linux and Windows
- Understand Input Output Management and use of Device Driver and Secondary Storage (Disk) Mechanism
- Analyze Security and Protection Mechanism in Operating System

Course Outcomes:

After learning, the course the students should be able to:

- Describe various generations of Operating System and functions of Operating System
- Describe the concept of program, process and thread and analyze various CPU Scheduling Algorithms and compare their performance
- Solve Inter Process Communication problems using Mathematical Equations by various methods
- Compare various Memory Management Schemes especially paging and Segmentation in Operating System and apply various Page Replacement Techniques
- Outline File Systems in Operating System like UNIX/Linux and Windows

UNIT I

Operating Systems Overview: Operating system functions, Operating system structure, Operating systems operations, Computing environments, Open-Source Operating Systems.
System Structures: Operating System Services, User and Operating-System Interface, systems calls, Types of System Calls, system programs, operating system structure, operating system debugging, System Boot.

UNIT II

Process Concept: Process scheduling, Operations on processes, Inter-process communication, Communication in client server systems.

Multithreaded Programming: Multithreading models, Thread libraries, Threading issues.

Process Scheduling: Basic concepts, Scheduling criteria, Scheduling algorithms, Multiple processor scheduling, Thread scheduling.

Inter-process Communication: Race conditions, Critical Regions, Mutual exclusion with busy waiting, Sleep and wakeup, Semaphores, Mutexes, Monitors, Message passing, Barriers, Classical IPC Problems - Dining philosophers problem, Readers and writers problem.

UNIT III

Memory-Management Strategies: Introduction, Swapping, Contiguous memory allocation, Paging, Segmentation.

Virtual Memory Management: Introduction, Demand paging, Copy on-write, Page replacement, Frame allocation, Thrashing, Memory-mapped files.

UNIT IV

Deadlocks: Resources, Conditions for resource deadlocks, Ostrich algorithm, Deadlock detection and recovery, Deadlock avoidance, Deadlock prevention.

File Systems: Files, Directories, File system implementation, management and optimization.

Secondary-Storage Structure: Overview of disk structure, and attachment, Disk scheduling, RAID structure, Stable storage implementation.

UNIT V

System Protection: Goals of protection, Principles and domain of protection, Access matrix, Access control, Revocation of access rights.

System Security: Introduction, Program threats, System and network threats, Implementing security defenses, SSL Certificates, Message digest algorithms.

Text Books:

- 1) Silberschatz A, Galvin P B, and Gagne G, Operating System Concepts, 9th edition, Wiley, 2013.
- 2) Tanenbaum A S, Modern Operating Systems, 3rd edition, Pearson Education, 2008. (for Interprocess Communication and File systems.)

Reference Books:

- 1) Dhamdhare D M, Operating Systems A Concept Based Approach, 3rd edition, Tata McGraw-Hill, 2012.
- 2) Stallings W, Operating Systems -Internals and Design Principles, 6th edition, Pearson Education, 2009
- 3) Nutt G, Operating Systems, 3rd edition, Pearson Education, 2004.

e-Resources:

- 1) <https://nptel.ac.in/courses/106/105/106105214/>

II Year – I Semester		L	T	P	C
		3	0	0	3
SOFTWARE ENGINEERING					

Course Objectives:

This course is designed to:

- Give exposure to phases of Software Development, common process models including Waterfall, and the Unified Process, and hands-on experience with elements of the agile process
- Give exposure to a variety of Software Engineering practices such as requirements analysis and specification, code analysis, code debugging, testing, traceability, and version control
- Give exposure to Software Design techniques

Course Outcomes:

Students taking this subject will gain software engineering skills in the following areas:

- Ability to transform an Object-Oriented Design into high quality, executable code
- Skills to design, implement, and execute test cases at the Unit and Integration level
- Compare conventional and agile software methods

UNIT I

The Nature of Software, The Unique Nature of WebApps, Software Engineering, The Software Process, Software Engineering Practice, Software Myths. A Generic Process Model, Process Assessment and Improvement, Prescriptive Process Models, Specialized Process Models, The Unified Process, Personal and Team Process Models, Process Technology.

UNIT II

Agility, Agility and the Cost of Change, Agile Process, Extreme Programming (XP), Other Agile Process Models, A Tool Set for the Agile Process, Software Engineering Knowledge, Core Principles, Principles That Guide Each Framework Activity, Requirements Engineering, Establishing the Groundwork, Eliciting Requirements, Developing Use Cases, Building the Requirements Model, Negotiating Requirements, Validating Requirements.

UNIT III

Requirements Analysis, Scenario-Based Modeling, UML Models That Supplement the Use Case, Data Modeling Concepts, Class-Based Modeling, Requirements Modeling Strategies, Flow-Oriented Modeling, Creating a Behavioral Model, Patterns for Requirements Modelling, Requirements Modeling for WebApps.

UNIT IV

Design within the Context of Software Engineering, The Design Process, Design Concepts, The Design Model, Software Architecture, Architectural Genres, Architectural Styles, Assessing Alternative Architectural Designs, Architectural Mapping Using Data Flow, Components, Designing Class-Based Components, Conducting Component-Level Design, Component-Level Design for WebApps, Designing Traditional Components, Component-Based Development.

UNIT V

The Golden Rules, User Interface Analysis and Design, Interface Analysis, Interface Design Steps, WebApp Interface Design, Design Evaluation, Elements of Software Quality Assurance, SQA Tasks, Goals & Metrics, Software Reliability.

A Strategic Approach to Software Testing, Strategic Issues, Test Strategies for Conventional Software, Test Strategies for Object-Oriented Software, Validation Testing, System Testing, The Art of Debugging, Software Testing Fundamentals, Internal and External Views of Testing, White-Box Testing, Basis Path Testing.

Text Books:

- 1) Software Engineering a practitioner's approach, Roger S. Pressman, Seventh Edition, McGraw Hill Higher Education.
- 2) Software Engineering, Ian Sommerville, Ninth Edition, Pearson.

Reference Books:

- 1) Software Engineering, A Precise Approach, PankajJalote, Wiley India, 2010.
- 2) Software Engineering, Ugrasen Suman, Cengage.

e-Resources:

- 1) <https://nptel.ac.in/courses/106/105/106105182/>

II Year – I Semester	L	T	P	C
	3	0	0	3
MATHEMATICAL FOUNDATIONS OF COMPUTER SCIENCE				

Course Objectives:

This course is designed to:

- To introduce the students to the topics and techniques of discrete methods and combinatorial reasoning
- To introduce a wide variety of applications. The algorithmic approach to the solution of problems is fundamental in discrete mathematics, and this approach reinforces the close ties between this discipline and the area of computer science

Course Outcomes:

At the end of the course student will be able to

- Demonstrate skills in solving mathematical problems
- Comprehend mathematical principles and logic
- Demonstrate knowledge of mathematical modeling and proficiency in using mathematical software
- Manipulate and analyze data numerically and/or graphically using appropriate Software
- Communicate effectively mathematical ideas/results verbally or in writing

UNIT I

Mathematical Logic: Propositional Calculus: Statements and Notations, Connectives, Well Formed Formulas, Truth Tables, Tautologies, Equivalence of Formulas, Duality Law, Tautological Implications, Normal Forms, Theory of Inference for Statement Calculus, Consistency of Premises, Indirect Method of Proof, Predicate Calculus: Predicates, Predicative Logic, Statement Functions, Variables and Quantifiers, Free and Bound Variables, Inference Theory for Predicate Calculus.

UNIT II

Set Theory: Sets: Operations on Sets, Principle of Inclusion-Exclusion, Relations: Properties, Operations, Partition and Covering, Transitive Closure, Equivalence, Compatibility and Partial Ordering, Hasse Diagrams, Functions: Bijective, Composition, Inverse, Permutation, and Recursive Functions, Lattice and its Properties, Algebraic Structures: Algebraic Systems, Properties, Semi Groups and Monoids, Group, Subgroup and Abelian Group, Homomorphism, Isomorphism.

UNIT III

Combinatorics: Basis of Counting, Permutations, Permutations with Repetitions, Circular and Restricted Permutations, Combinations, Restricted Combinations, Binomial and Multinomial Coefficients and Theorems, Number Theory: Properties of Integers, Division Theorem,

Greatest Common Divisor, Euclidean Algorithm, Least Common Multiple, Testing for Prime Numbers, The Fundamental Theorem of Arithmetic, Modular Arithmetic, Fermat's and Euler's Theorems

UNIT IV

Recurrence Relations: Generating Functions, Function of Sequences, Partial Fractions, Calculating Coefficient of Generating Functions, Recurrence Relations, Formulation as Recurrence Relations, Solving Recurrence Relations by Substitution and Generating Functions, Method of Characteristic Roots, Solving Inhomogeneous Recurrence Relations

UNIT V

Graph Theory: Basic Concepts, Graph Theory and its Applications, Sub graphs, Graph Representations: Adjacency and Incidence Matrices, Isomorphic Graphs, Paths and Circuits, Eulerian and Hamiltonian Graphs, Multigraphs, Bipartite and Planar Graphs, Euler's Theorem, Graph Colouring and Covering, Chromatic Number, Spanning Trees, Prim's and Kruskal's Algorithms, BFS and DFS Spanning Trees.

Text Books:

- 1) Discrete Mathematical Structures with Applications to Computer Science, J. P. Tremblay and P. Manohar, Tata McGraw Hill.
- 2) Elements of Discrete Mathematics-A Computer Oriented Approach, C. L. Liu and D. P. Mohapatra, 3rd Edition, Tata McGraw Hill.

Reference Books:

- 1) Discrete Mathematics for Computer Scientists and Mathematicians, J. L. Mott, A. Kandel and T. P. Baker, 2nd Edition, Prentice Hall of India.
- 2) Discrete Mathematical Structures, Bernard Kolman, Robert C. Busby and Sharon Cutler Ross, PHI.
- 3) Discrete Mathematics and its Applications with Combinatorics and Graph Theory, K. H. Rosen, 7th Edition, Tata McGraw Hill.

e-Resources:

- 1) <https://nptel.ac.in/courses/106/106/106106094/>

II Year – I Semester		L	T	P	C
		0	0	3	1.5
OBJECT ORIENTED PROGRAMMING THROUGH C++ LAB					

Course Objectives:

The objective of this lab is to

- Demonstrate procedural and object oriented paradigm with concepts of streams, classes, functions, data and objects.
- Understand dynamic memory management techniques using pointers, constructors, destructors, etc
- Demonstrate the concept of function overloading, operator overloading, virtual functions and polymorphism, inheritance.

Course Outcomes:

By the end of this lab the student is able to

- Apply the various OOPs concepts with the help of programs.

Exercise -1 (Classes Objects)

Create a Distance class with:

- feet and inches as data members
 - member function to input distance
 - member function to output distance
 - member function to add two distance objects
1. Write a main function to create objects of DISTANCE class. Input two distances and output the sum.
 2. Write a C++ Program to illustrate the use of Constructors and Destructors (use the above program.)
 3. Write a program for illustrating function overloading in adding the distance between objects (use the above problem)

Exercise – 2 (Access)

Write a program for illustrating Access Specifiers public, private, protected

1. Write a program implementing Friend Function
2. Write a program to illustrate this pointer
3. Write a Program to illustrate pointer to a class

Exercise -3 (Operator Overloading)

1. Write a program to Overload Unary, and Binary Operators as Member Function, and Non Member Function.
 1. Unary operator as member function
 2. Binary operator as non member function
2. Write a c ++ program to implement the overloading assignment = operator

Exercise -4 (Inheritance)

1. Write C++ Programs and incorporating various forms of Inheritance
 - i) Single Inheritance
 - ii) Hierarchical Inheritance
 - iii) Multiple Inheritances
 - iv) Multi-level inheritance
 - v) Hybrid inheritance
2. Also illustrate the order of execution of constructors and destructors in inheritance

Exercise -5(Templates, Exception Handling)

1. a)Write a C++ Program to illustrate template class
2. b)Write a Program to illustrate member function templates
3. c) Write a Program for Exception Handling Divide by zero
4. d)Write a Program to rethrow an Exception

Exercise -6

1. Write a C++ program illustrating user defined string processing functions using pointers (string length, string copy, string concatenation)
2. Write a C++ program illustrating Virtual classes & virtual functions.
3. Write C++ program that implement Bubble sort, to sort a given list of integers in ascending order

II Year – I Semester	L	T	P	C
	0	0	3	1.5
OPERATING SYSTEM LAB				

Course Objectives:

- To understand the design aspects of operating system
- To study the process management concepts & Techniques
- To study the storage management concepts
- To familiarize students with the Linux environment
- To learn the fundamentals of shell scripting/programming

Course Outcomes:

- To use Unix utilities and perform basic shell control of the utilities
- To use the Unix file system and file access control
- To use of an operating system to develop software
- Students will be able to use Linux environment efficiently
- Solve problems using bash for shell scripting

- 1) a) Study of Unix/Linux general purpose utility command list: man, who, cat, cd, cp, ps, ls, mv, rm, mkdir, rmdir, echo, more, date, time, kill, history, chmod, chown, finger, pwd, cal, logout, shutdown.
b) Study of vi editor
c) Study of Bash shell, Bourne shell and C shell in Unix/Linux operating system
d) Study of Unix/Linux file system (tree structure)
e) Study of .bashrc, /etc/bashrc and Environment variables.
- 2) Write a C program that makes a copy of a file using standard I/O, and system calls
- 3) Write a C program to emulate the UNIX ls -l command.
- 4) Write a C program that illustrates how to execute two commands concurrently with a command pipe. Ex: - ls -l | sort
- 5) Simulate the following CPU scheduling algorithms:
(a) Round Robin (b) SJF (c) FCFS (d) Priority
- 6) Multiprogramming-Memory management-Implementation of fork (), wait (), exec() and exit (), System calls
- 7) Simulate the following:
a) Multiprogramming with a fixed number of tasks (MFT)
b) Multiprogramming with a variable number of tasks (MVT)
- 8) Simulate Bankers Algorithm for Dead Lock Avoidance
- 9) Simulate Bankers Algorithm for Dead Lock Prevention.
- 10) Simulate the following page replacement algorithms:
a) FIFO b) LRU c) LFU
- 11) Simulate the following File allocation strategies
(a) Sequenced (b) Indexed (c) Linked
- 12) Write a C program that illustrates two processes communicating using shared memory
- 13) Write a C program to simulate producer and consumer problem using semaphores

14) Write C program to create a thread using pthreads library and let it run its function.

15) Write a C program to illustrate concurrent execution of threads using pthreads library.

II Year – I Semester		L	T	P	C
		0	0	3	1.5
SOFTWARE ENGINEERING LAB					

Course Objectives:

The objective of this lab is to acquire the generic software development skill through various stages of software life cycle and also to ensure the quality of software through software development with various protocol based environment

Course Outcomes:

- By the end of this lab the student is able to elicit, analyze and specify software requirements through a productive working relationship with various stakeholders of the project
- prepare SRS document, design document, test cases and software configuration management and risk management related document.
- develop function oriented and object oriented software design using tools like rational rose.
- use modern engineering tools necessary for software project management, estimations, time management and software reuse
- generate test cases for software testing

1) Perform the following, for the following experiments:

- i. Do the Requirement Analysis and Prepare SRS
- ii. Draw E-R diagrams, DFD, CFD and structured charts for the project.

- 1) Course Registration System
- 2) Students Marks Analyzing System
- 3) Online Ticket Reservation System
- 4) Stock Maintenance

5) Consider any application, using COCOMO model, estimate the effort.

6) Consider any application, Calculate effort using FP oriented estimation model.

7) Draw the UML Diagrams for the problem 1,2, 3, 4.

8) Design the test cases for e-Commerce application (Flipcart, Amazon)

9) Design the test cases for a Mobile Application (Consider any example from Appstore)

10) Design and Implement ATM system through UML Diagrams.

II Year - I Semester	Skill Oriented Course- I	L	T	P	C
		0	0	4	2
APPLICATIONS OF PYTHON- NumPy					

Course Objectives:

The objective of this lab is to acquire programming skills in Python package NumPy and perform mathematical and statistical operations.

Course Outcomes:

By the end of this lab the student is able to

- Explain how data is collected, managed and stored for processing
- Understand the workings of various numerical techniques, different descriptive measures of Statistics, correlation and regression to solve the engineering problems
- Understand how to apply some linear algebra operations to n-dimensional arrays
- Use NumPy perform common data wrangling and computational tasks in Python.

Perform the following:

- 1) NumPy Installation using different scientific python distributions(Anaconda, Python(x,y), WinPython, Pyzo)
- 2) NumPy Basics (np.array, np.arange, np.linspace, np.zeros, np.ones, np.random.random, np.empty)
- 3) Arrays (array.shape, len(array), array.ndim, array.dtype, array.astype(type), type(array))
- 4) Array Manipulation (np.append, np.insert, np.resize, np.delete, np.concatenate, np.vstack, np.hstack)
- 5) Mathematical Operations(np.add, np.subtract, np.divide, np.multiply, np.sqrt, np.sin, np.cos, np.log, np.dot, np.roots) , Statistical Operations(np.mean, np.median, np.std, array.corrcoef())
- 6) NumPy data types
- 7) NumPy ndarray
- 8) NumPy String Operations
- 9) NumPy Financial functions
- 10) NumPy Functional Programming

II Year - I Semester	L	T	P	C
	2	0	0	0
Essence of Indian traditional Knowledge				

Course Objectives: The course should enable the students to:

- Understand the concept of Traditional knowledge and its importance
- Know the need and importance of protecting traditional knowledge.
- Know the various enactments related to the protection of traditional knowledge.
- Understand the concepts of Intellectual property to protect the traditional knowledge.

Course Outcomes: After successful completion of this course, students will be able to:

- Understand and elucidate the basic knowledge of traditional knowledge to Understand develop the physical and social changes on traditional knowledge system.
- Recognize the role of government on traditional knowledge to measure itsApply impact on global economy.
- Illustrate the rules of biological diversity and geographical indicators for theUnderstand protection of traditional knowledge bill.
- Evaluate the legal mechanism of traditional knowledge protection to show theRemember difference between IPR and non-IPR system.
- Examine the sustainability and development of environment for standardizingApply the food security and traditional knowledge of the country.

UNIT I: INTRODUCTION TO TRADITIONAL KNOWLEDGE

Define traditional knowledge, nature and characteristics, scope and importance, kinds of traditional knowledge, the physical and social contexts in which traditional knowledge develop, the historical impact of social change on traditional knowledge systems. Indigenous Knowledge (IK), characteristics, traditional knowledge vis-à-vis indigenous knowledge, traditional knowledge Vs western knowledge traditional knowledge vis-à-vis formal knowledge

UNIT II: PROTECTION OF TRADITIONAL KNOWLEDGE

Protection of traditional knowledge: The need for protecting traditional knowledge Significance of TKProtection, value of TK in global economy, Role of Government to harness TK

UNIT III: LEGAL FRAMEWORK AND TK

A: The Scheduled Tribes and Other Traditional Forest Dwellers (Recognition of Forest Rights) Act, 2006, Plant Varieties Protection and Farmer's Rights Act, 2001 (PPVFR Act);
 B: The Biological Diversity Act 2002 and Rules 2004, the protection of traditional knowledge bill, 2016. Geographical indicators act 2003.

UNIT IV: TRADITIONAL KNOWLEDGE AND INTELLECTUAL PROPERTY

Systems of traditional knowledge protection, Legal concepts for the protection of traditional knowledge, Certain non IPR mechanisms of traditional knowledge protection, Patents and traditional knowledge, Strategies to increase protection of traditional knowledge, global legal FORA for increasing protection of Indian Traditional Knowledge.

UNIT V: TRADITIONAL KNOWLEDGE IN DIFFERENT SECTORS

Traditional knowledge and engineering, Traditional medicine system, TK and biotechnology, TK in agriculture, Traditional societies depend on it for their food and healthcare needs, Importance of conservation and sustainable development of environment, Management of biodiversity, Food security of the country and protection of TK. 139

Text Books:

1. Traditional Knowledge System in India, by Amit Jha, 2009.
2. Traditional Knowledge System and Technology in India by Basanta Kumar Mohanta and Vipin KumaSingh, Pratibha Prakashan 2012.

Reference Books:

1. Traditional Knowledge System in India by Amit Jha Atlantic publishers, 2002.
2. "Knowledge Traditions and Practices of India" Kapil Kapoor¹, Michel Danino²

II Year – II Semester	L	T	P	C
	3	0	0	3
PROBABILITY AND STATISTICS				

Course Objectives:

- To familiarize the students with the foundations of probability and statistical methods
- To impart probability concepts and statistical methods in various applications
Engineering

Course Outcomes:

Upon successful completion of this course, the student should be able to

- Classify the concepts of data science and its importance (L4) or (L2)
- Interpret the association of characteristics and through correlation and regression tools(L4)
- Make use of the concepts of probability and their applications (L3)
- Apply discrete and continuous probability distributions (L3)
- Design the components of a classical hypothesis test (L6)
- Infer the statistical inferential methods based on small and large sampling tests (L4)

UNIT I

Descriptive statistics and methods for data science: Data science – Statistics Introduction – Population vs Sample – Collection of data – primary and secondary data – Type of variable: dependent and independent Categorical and Continuous variables – Data visualization – Measures of Central tendency – Measures of Variability (spread or variance) – Skewness Kurtosis.

UNIT II

Correlation and Curve fitting: Correlation – correlation coefficient – rank correlation – regression coefficients and properties – regression lines – Method of least squares – Straight line – parabola – Exponential – Power curves.

UNIT III

Probability and Distributions: Probability – Conditional probability and Baye’s theorem – Random variables – Discrete and Continuous random variables – Distribution function – Mathematical Expectation and Variance – Binomial, Poisson, Uniform and Normal distributions.

UNIT IV

Sampling Theory: Introduction – Population and samples – Sampling distribution of Means and Variance (definition only) – Central limit theorem (without proof) – Introduction to t, χ^2 and F-distributions – Point and Interval estimations – Maximum error of estimate.

UNIT V

Tests of Hypothesis: Introduction – Hypothesis – Null and Alternative Hypothesis – Type I and Type II errors – Level of significance – One tail and two-tail tests – Tests concerning one mean and two means (Large and Small samples) – Tests on proportions.

Text Books:

- 1) Miller and Freund's, Probability and Statistics for Engineers, 7/e, Pearson, 2008.
- 2) S. C. Gupta and V.K. Kapoor, Fundamentals of Mathematical Statistics, 11/e, Sultan Chand & Sons Publications, 2012.

Reference Books:

- 1) Shron L. Myers, Keying Ye, Ronald E Walpole, Probability and Statistics Engineers and the Scientists, 8th Edition, Pearson 2007.
- 2) Jay I. Devore, Probability and Statistics for Engineering and the Sciences, 8th Edition, Cengage.
- 3) Sheldon M. Ross, Introduction to probability and statistics Engineers and the Scientists, 4th Edition, Academic Foundation, 2011.
- 4) Johannes Ledolter and Robert V. Hogg, Applied statistics for Engineers and Physical Scientists, 3rd Edition, Pearson, 2010.

II Year – II Semester		L	T	P	C
		3	0	0	3
DATABASE MANAGEMENT SYSTEMS					

Course Objectives:

- To introduce about database management systems
- To give a good formal foundation on the relational model of data and usage of Relational Algebra
- To introduce the concepts of basic SQL as a universal Database language
- To demonstrate the principles behind systematic database design approaches by covering conceptual design, logical design through normalization
- To provide an overview of physical design of a database system, by discussing Database indexing techniques and storage techniques

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

- Describe a relational database and object-oriented database
- Create, maintain and manipulate a relational database using SQL
- Describe ER model and normalization for database design
- Examine issues in data storage and query processing and can formulate appropriate solutions
- Outline the role and issues in management of data such as efficiency, privacy, security, ethical responsibility, and strategic advantage

UNIT I

Introduction: Database system, Characteristics (Database Vs File System), Database Users (Actors on Scene, Workers behind the scene), Advantages of Database systems, Database applications. Brief introduction of different Data Models; Concepts of Schema, Instance and data independence; Three tier schema architecture for data independence; Database system structure, environment, Centralized and Client Server architecture for the database.

UNIT II

Relational Model: Introduction to relational model, concepts of domain, attribute, tuple, relation, importance of null values, constraints (Domain, Key constraints, integrity constraints) and their importance BASIC SQL: Simple Database schema, data types, table definitions (create, alter), different DML operations (insert, delete, update), basic SQL querying (select and project) using where clause, arithmetic & logical operations, SQL functions (Date and Time, Numeric, String conversion).

UNIT III

Entity Relationship Model: Introduction, Representation of entities, attributes, entity set, relationship, relationship set, constraints, sub classes, super class, inheritance, specialization, generalization using ER Diagrams. SQL: Creating tables with relationship, implementation of key and integrity constraints, nested queries, sub queries, grouping, aggregation, ordering, implementation of different types of joins, view (updatable and non-updatable), relational set operations.

UNIT IV

Schema Refinement (Normalization): Purpose of Normalization or schema refinement, concept of functional dependency, normal forms based on functional dependency(1NF, 2NF and 3 NF), concept of surrogate key, Boyce-codd normal form(BCNF), Lossless join and dependency preserving decomposition, Fourth normal form(4NF), Fifth Normal Form (5NF).

UNIT V

Transaction Concept: Transaction State, Implementation of Atomicity and Durability, Concurrent Executions, Serializability, Recoverability, Implementation of Isolation, Testing for Serializability.

Indexing Techniques: B+ Trees: Search, Insert, Delete algorithms, File Organization and Indexing Indexes and Performance Tuning

Text Books:

- 1) Database Management Systems, 3/e, Raghurama Krishnan, Johannes Gehrke, TMH
- 2) Database System Concepts,5/e, Silberschatz, Korth, TMH

Reference Books:

- 1) Introduction to Database Systems, 8/e C J Date, PEA.
- 2) Database Management System, 6/e Ramez Elmasri, Shamkant B. Navathe, PEA
- 3) Database Principles Fundamentals of Design Implementation and Management, CorlosCoronel, Steven Morris, Peter Robb, Cengage Learning.

e-Resources:

- 1) <https://nptel.ac.in/courses/106/105/106105175/>
- 2) <https://www.geeksforgeeks.org/introduction-to-nosql/>

II Year – II Semester		L	T	P	C
		3	0	0	3
FORMAL LANGUAGES AND AUTOMATA THEORY					

Course Objectives:

- To learn fundamentals of Regular and Context Free Grammars and Languages
- To understand the relation between Regular Language and Finite Automata and machines
- To learn how to design Automata's and machines as Acceptors, Verifiers and Translators
- To understand the relation between Contexts free Languages, PDA and TM
- To learn how to design PDA as acceptor and TM as Calculators

Course Outcomes: By the end of the course students can

- Classify machines by their power to recognize languages.
- Summarize language classes & grammars relationship among them with the help of Chomsky hierarchy
- Employ finite state machines to solve problems in computing
- Illustrate deterministic and non-deterministic machines
- Quote the hierarchy of problems arising in the computer science

UNIT I

Finite Automata: Need of Automata theory, Central Concepts of Automata Theory, Automation, Finite Automata, Transition Systems, Acceptance of a String, DFA, Design of DFAs, NFA, Design of NFA, Equivalence of DFA and NFA, Conversion of NFA into DFA, Finite Automata with ϵ -Transitions, Minimization of Finite Automata, Finite Automata with output-Mealy and Moore Machines, Applications and Limitation of Finite Automata.

UNIT II

Regular Expressions, Regular Sets, Identity Rules, Equivalence of two RE, Manipulations of REs, Finite Automata and Regular Expressions, Inter Conversion, Equivalence between FA and RE, Pumping Lemma of Regular Sets, Closure Properties of Regular Sets, Grammars, Classification of Grammars, Chomsky Hierarchy Theorem, Right and Left Linear Regular Grammars, Equivalence between RG and FA, Inter Conversion.

UNIT III

Formal Languages, Context Free Grammar, Leftmost and Rightmost Derivations, Parse Trees, Ambiguous Grammars, Simplification of Context Free Grammars-Elimination of Useless Symbols, ϵ -Productions and Unit Productions, Normal Forms-Chomsky Normal Form and Greibach Normal Form, Pumping Lemma, Closure Properties, Applications of Context Free Grammars.

UNIT IV

Pushdown Automata, Definition, Model, Graphical Notation, Instantaneous Description, Language Acceptance of Pushdown Automata, Design of Pushdown Automata, Deterministic and Non – Deterministic Pushdown Automata, Equivalence of Pushdown Automata and Context Free Grammars, Conversion, Two Stack Pushdown Automata, Application of Pushdown Automata.

UNIT V

Turning Machine: Definition, Model, Representation of TMs-Instantaneous Descriptions, Transition Tables and Transition Diagrams, Language of a TM, Design of TMs, Universal and Restricted TM, Decidable and Un-decidable Problems, Halting Problem of TMs, Post's Correspondence Problem,

Text Books:

- 1) Introduction to Automata Theory, Languages and Computation, J. E. Hopcroft, R. Motwani and J. D. Ullman, 3rd Edition, Pearson, 2008
- 2) Theory of Computer Science-Automata, Languages and Computation, K. L. P. Mishra and N. Chandrasekharan, 3rd Edition, PHI, 2007

Reference Books:

- 1) Elements of Theory of Computation, Lewis H.P. & Papadimition C.H., Pearson /PHI
- 2) Theory of Computation, V. Kulkarni, Oxford University Press, 2013
- 3) Theory of Automata, Languages and Computation, Rajendra Kumar, McGraw Hill, 2014

e-Resources:

- 1) <https://nptel.ac.in/courses/106/104/106104028/>

II Year – II Semester	L	T	P	C
	3	0	0	3
JAVA PROGRAMMING				

Course Objectives: The learning objectives of this course are:

- To identify Java language components and how they work together in applications
- To learn the fundamentals of object-oriented programming in Java, including defining classes, invoking methods, using class libraries.
- To learn how to extend Java classes with inheritance and dynamic binding and how to use exception handling in Java applications
- To understand how to design applications with threads in Java
- To understand how to use Java APIs for program development

Course Outcomes: By the end of the course, the student will be

- Able to realize the concept of Object Oriented Programming & Java Programming Constructs
- Able to describe the basic concepts of Java such as operators, classes, objects, inheritance, packages, Enumeration and various keywords
- Apply the concept of exception handling and Input/ Output operations
- Able to design the applications of Java & Java applet
- Able to Analyze & Design the concept of Event Handling and Abstract Window Toolkit

UNIT I

Program Structure in Java: Introduction, Writing Simple Java Programs, Elements or Tokens in Java Programs, Java Statements, Command Line Arguments, User Input to Programs, Escape Sequences Comments, Programming Style.

Data Types, Variables, and Operators: Introduction, Data Types in Java, Declaration of Variables, Data Types, Type Casting, Scope of Variable Identifier, Literal Constants, Symbolic Constants, Formatted Output with printf() Method, Static Variables and Methods, Attribute Final, Introduction to Operators, Precedence and Associativity of Operators, Assignment Operator (=), Basic Arithmetic Operators, Increment (++) and Decrement (- -) Operators, Ternary Operator, Relational Operators, Boolean Logical Operators, Bitwise Logical Operators.

Control Statements: Introduction, if Expression, Nested if Expressions, if-else Expressions, Ternary Operator?., Switch Statement, Iteration Statements, while Expression, do-while Loop, for Loop, Nested for Loop, For-Each for Loop, Break Statement, Continue Statement.

UNIT II

Classes and Objects: Introduction, Class Declaration and Modifiers, Class Members, Declaration of Class Objects, Assigning One Object to Another, Access Control for Class Members, Accessing Private Members of Class, Constructor Methods for Class, Overloaded Constructor Methods, Nested Classes, Final Class and Methods, Passing Arguments by Value and by Reference, Keyword this.

Methods: Introduction, Defining Methods, Overloaded Methods, Overloaded Constructor Methods, Class Objects as Parameters in Methods, Access Control, Recursive Methods, Nesting of Methods, Overriding Methods, Attributes Final and Static.

UNIT III

Arrays: Introduction, Declaration and Initialization of Arrays, Storage of Array in Computer Memory, Accessing Elements of Arrays, Operations on Array Elements, Assigning Array to Another Array, Dynamic Change of Array Size, Sorting of Arrays, Search for Values in Arrays, Class Arrays, Two-dimensional Arrays, Arrays of Varying Lengths, Three-dimensional Arrays, Arrays as Vectors.

Inheritance: Introduction, Process of Inheritance, Types of Inheritances, Universal Super

Class-Object Class, Inhibiting Inheritance of Class Using Final, Access Control and Inheritance, Multilevel Inheritance, Application of Keyword Super, Constructor Method and Inheritance, Method Overriding, Dynamic Method Dispatch, Abstract Classes, Interfaces and Inheritance.

Interfaces: Introduction, Declaration of Interface, Implementation of Interface, Multiple Interfaces, Nested Interfaces, Inheritance of Interfaces, Default Methods in Interfaces, Static Methods in Interface, Functional Interfaces, Annotations.

UNIT IV

Packages and Java Library: Introduction, Defining Package, Importing Packages and Classes into Programs, Path and Class Path, Access Control, Packages in Java SE, Java.lang Package and its Classes, Class Object, Enumeration, class Math, Wrapper Classes, Auto-boxing and Auto-unboxing, Java util Classes and Interfaces, Formatter Class, Random Class, Time Package, Class Instant (java.time.Instant), Formatting for Date/Time in Java, Temporal Adjusters Class, Temporal Adjusters Class.

Exception Handling: Introduction, Hierarchy of Standard Exception Classes, Keywords throws and throw, try, catch, and finally Blocks, Multiple Catch Clauses, Class Throwable, Unchecked Exceptions, Checked Exceptions, try-with-resources, Catching Subclass Exception, Custom Exceptions, Nested try and catch Blocks, Rethrowing Exception, Throws Clause.

UNIT V

String Handling in Java: Introduction, Interface Char Sequence, Class String, Methods for Extracting Characters from Strings, Methods for Comparison of Strings, Methods for Modifying Strings, Methods for Searching Strings, Data Conversion and Miscellaneous Methods, Class String Buffer, Class String Builder.

Multithreaded Programming: Introduction, Need for Multiple Threads Multithreaded Programming for Multi-core Processor, Thread Class, Main Thread- Creation of New Threads, Thread States, Thread Priority-Synchronization, Deadlock and Race Situations, Inter-thread Communication - Suspending, Resuming, and Stopping of Threads.

Java Database Connectivity: Introduction, JDBC Architecture, Installing MySQL and MySQL Connector/J, JDBC Environment Setup, Establishing JDBC Database Connections, ResultSet Interface, Creating JDBC Application, JDBC Batch Processing, JDBC Transaction Management

Text Books:

- 1) JAVA one step ahead, Anitha Seth, B.L.Juneja, Oxford.
- 2) The complete Reference Java, 8th edition, Herbert Schildt, TMH.

References Books:

- 1) Introduction to java programming, 7th edition by Y Daniel Liang, Pearson
- 2) Murach's Java Programming, Joel Murach

e-Resources:

- 1) <https://nptel.ac.in/courses/106/105/106105191/>
- 2) https://www.w3schools.com/java/java_data_types.asp

II Year – II Semester	L	T	P	C
	3	0	0	3
MANAGERIAL ECONOMICS AND FINANCIAL ACCOUNTANCY				

Course Objectives:

- The Learning objectives of this paper are to understand the concept and nature of Managerial Economics and its relationship with other disciplines and also to understand the Concept of Demand and Demand forecasting
- To familiarize about the Production function, Input Output relationship, Cost-Output relationship and Cost-Volume-Profit Analysis
- To understand the nature of markets, Methods of Pricing in the different market structures and to know the different forms of Business organization and the concept of Business Cycles
- To learn different Accounting Systems, preparation of Financial Statement and uses of different tools for performance evaluation
- Finally, it is also to understand the concept of Capital, Capital Budgeting and the techniques used to evaluate Capital Budgeting proposals

Course Outcomes:

- The Learner is equipped with the knowledge of estimating the Demand and demand elasticities for a product
- The knowledge of understanding of the Input-Output-Cost relationships and estimation of the least cost combination of inputs
- The pupil is also ready to understand the nature of different markets and Price Output determination under various market conditions and also to have the knowledge of different Business Units
- The Learner is able to prepare Financial Statements and the usage of various Accounting tools for Analysis
- The Learner can able to evaluate various investment project proposals with the help of capital budgeting techniques for decision making

UNIT I

Introduction to Managerial Economics and demand Analysis: Definition of Managerial Economics – Scope of Managerial Economics and its relationship with other subjects – Concept of Demand, Types of Demand, Determinants of Demand- Demand schedule, Demand curve, Law of Demand and its limitations- Elasticity of Demand, Types of Elasticity of Demand and Measurement- Demand forecasting and Methods of forecasting, Concept of Supply and Law of Supply.

UNIT II

Theories of Production and Cost Analyses: Theories of Production function- Law of Variable proportions-Isoquants and Isocosts and choice of least cost factor combination-Concepts of Returns to scale and Economies of scale-Different cost concepts: opportunity costs, explicit and implicit costs- Fixed costs, Variable Costs and Total costs –Cost –Volume-Profit analysis-Determination of Breakeven point(problems)-Managerial significance and limitations of Breakeven point.

UNIT III

Introduction to Markets, Theories of the Firm & Pricing Policies: Market Structures: Perfect Competition, Monopoly, Monopolistic competition and Oligopoly – Features – Price and Output Methods of Pricing: Average cost pricing, Limit Pricing, Market Skimming Pricing, Internet Pricing: (Flat Rate Pricing, Usage sensitive pricing) and Priority Pricing, Business Cycles : Meaning and Features – Phases of a Business Cycle. Features and Evaluation of Sole Trader, Partnership, Joint Stock Company – State/Public Enterprises and their forms.

UNIT IV

Introduction to Accounting & Financing Analysis: Introduction to Double Entry System, Journal, Ledger, Trail Balance and Preparation of Final Accounts with adjustments – Preparation of Financial Statements-Analysis and Interpretation of Financial Statements-Ratio Analysis – Preparation of Fundsflow and cash flow analysis (Problems)

UNIT V

Capital and Capital Budgeting: Capital Budgeting: Meaning of Capital-Capitalization-Meaning of Capital Budgeting-Time value of money- Methods of appraising Project profitability: Traditional Methods (pay back period, accounting rate of return) and modern methods(Discounted cash flow method, Net Present Value method, Internal Rate of Return Method and Profitability Index)

Text Books:

- 1) A R Aryasri, Managerial Economics and Financial Analysis, The McGraw – Hill companies.

Reference Books:

- 1) Varshney R.L, K.L Maheswari, Managerial Economics, S. Chand & Company Ltd.
- 2) JL Pappas and EF Brigham, Managerial Economics, Holt, R & W; New edition edition
- 3) N.P Srinivasn and M. SakthivelMurugan, Accounting for Management, S. Chand & CompanyLtd.
- 4) MaheswariS.N,AnIntroduction to Accountancy, Vikas Publishing House Pvt Ltd
- 5) I.M Pandey, Financial Management , Vikas Publishing House Pvt Ltd
- 6) V. Maheswari, Managerial Economics, S. Chand & Company Ltd.

II Year – II Semester		L	T	P	C
		0	0	2	1
DATABASE MANAGEMENT SYSTEMS LAB					

Course Objectives:

This Course will enable students to

- Populate and query a database using SQL DDL/DML Commands
- Declare and enforce integrity constraints on a database
- Writing Queries using advanced concepts of SQL
- Programming PL/SQL including procedures, functions, cursors and triggers

Course Outcomes:

At the end of the course the student will be able to:

- Utilize SQL to execute queries for creating database and performing data manipulation operations
- Examine integrity constraints to build efficient databases
- Apply Queries using Advanced Concepts of SQL
- Build PL/SQL programs including stored procedures, functions, cursors and triggers

List of Exercises:

1. Creation, altering and dropping of tables and inserting rows into a table (use constraints while creating tables) examples using SELECT command.
2. Queries (along with sub Queries) using ANY, ALL, IN, EXISTS, NOTEXISTS, UNION, INTERSET, Constraints. Example:- Select the roll number and name of the student who secured fourth rank in the class.
3. Queries using Aggregate functions (COUNT, SUM, AVG, MAX and MIN), GROUP BY, HAVING and Creation and dropping of Views.
4. Queries using Conversion functions (to_char, to_number and to_date), string functions (Concatenation, lpad, rpad, ltrim, rtrim, lower, upper, initcap, length, substr and instr), date functions (Sysdate, next_day, add_months, last_day, months_between, least, greatest, trunc, round, to_char, to_date)
5.
 - i. Create a simple PL/SQL program which includes declaration section, executable section and exception –Handling section (Ex. Student marks can be selected from the table and printed for those who secured first class and an exception can be raised if no records were found)
 - ii. Insert data into student table and use COMMIT, ROLLBACK and SAVEPOINT in PL/SQL block.
6. Develop a program that includes the features NESTED IF, CASE and CASE expression. The program can be extended using the NULLIF and COALESCE functions.
7. Program development using WHILE LOOPS, numeric FOR LOOPS, nested loops using ERROR Handling, BUILT –IN Exceptions, USE defined Exceptions, RAISE-APPLICATION ERROR.
8. Programs development using creation of procedures, passing parameters IN and OUT of PROCEDURES.
9. Program development using creation of stored functions, invoke functions in SQL Statements and write complex functions.

10. Develop programs using features parameters in a CURSOR, FOR UPDATE CURSOR, WHERE CURRENT of clause and CURSOR variables.
11. Develop Programs using BEFORE and AFTER Triggers, Row and Statement Triggers and INSTEAD OF Triggers
12. Create a table and perform the search operation on table using indexing and non-indexing techniques.

Text Books/Suggested Reading:

- 1) Oracle: The Complete Reference by Oracle Press
- 2) Nilesh Shah, "Database Systems Using Oracle", PHI, 2007
- 3) Rick F Vander Lans, "Introduction to SQL", Fourth Edition, Pearson Education, 2007

II Year – II Semester	L	T	P	C
	0	1	2	2
R PROGRAMMING LAB				

COURSE OBJECTIVES:

- To learn statistical programming, computation, graphics, and modeling,
- To learn Writing functions and use R in an efficient way,
- To learn about basic types of statistical models

COURSE OUTCOMES:

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

- Access online resources for R and import new function packages into the R workspace
 - Import, review, manipulate and summarize data-sets in R
 - Explore data-sets to create testable hypotheses and identify appropriate statistical tests
 - Perform appropriate statistical tests using R
 - Create and edit visualizations with R
- 1) Write a R program to take input from the user (name and age) and display the values. Also print the version of R installation.
 - 2) Write a R program to get the details of the objects in memory.
 - 3) Write a R program to create a sequence of numbers from 20 to 50 and find the mean of numbers from 20 to 60 and sum of numbers from 51 to 91.
 - 4) Write a R program to create a simple bar plot of five subjects marks.
 - 5) Write a R program to get the unique elements of a given string and unique numbers of vector.
 - 6) Write a R program to create three vectors a,b,c with 3 integers. Combine the three vectors to become a 3×3 matrix where each column represents a vector. Print the content of the matrix.
 - 7) Write a R program to create a 5 x 4 matrix , 3 x 3 matrix with labels and fill the matrix by rows and 2 × 2 matrix with labels and fill the matrix by columns.
 - 8) Write a R program to combine three arrays so that the first row of the first array is followed by the first row of the second array and then first row of the third array.
 - 9) Write a R program to create a two-dimensional 5x3 array of sequence of even integers greater than 50.
 - 10) Write a R program to create an array using four given columns, three given rows, and two given tables and display the content of the array.
 - 11) Write a R program to create an empty data frame.
 - 12) Write a R program to create a data frame from four given vectors.
 - 13) Write a R program to create a data frame using two given vectors and display the duplicated elements and unique rows of the said data frame.
 - 14) Write a R program to save the information of a data frame in a file and display the information of the file.
 - 15) Write a R program to create a matrix from a list of given vectors.
 - 16) Write a R program to concatenate two given matrices of same column but different rows.
 - 17) Write a R program to find row and column index of maximum and minimum value in a given matrix.

- 18) Write a R program to append value to a given empty vector.
- 19) Write a R program to multiply two vectors of integers type and length 3.
- 20) Write a R program to find Sum, Mean and Product of a Vector, ignore element like NA or NaN.
- 21) Write a R program to list containing a vector, a matrix and a list and give names to the elements in the list.
- 22) Write a R program to create a list containing a vector, a matrix and a list and give names to the elements in the list. Access the first and second element of the list.
- 23) Write a R program to create a list containing a vector, a matrix and a list and remove the second element.
- 24) Write a R program to select second element of a given nested list.
- 25) Write a R program to merge two given lists into one list.
- 26) Write a R program to create a list named s containing sequence of 15 capital letters, starting from 'E'.
- 27) Write a R program to assign new names "a", "b" and "c" to the elements of a given list.
- 28) Write a R program to find the levels of factor of a given vector.
- 29) Write a R program to create an ordered factor from data consisting of the names of months.
- 30) Write a R program to concatenate two given factor in a single factor.

II Year – II Semester		L	T	P	C
		0	0	3	1.5
JAVA PROGRAMMING LAB					

Course Objectives:

The aim of this lab is to

- Practice programming in the Java
- Gain knowledge of object-oriented paradigm in the Java programming language
- Learn use of Java in a variety of technologies and on different platforms

Course Outcomes:

By the end of the course student will be able to write java program for

- Evaluate default value of all primitive data type, Operations, Expressions, Control-flow, Strings
- Determine Class, Objects, Methods, Inheritance, Exception, Runtime Polymorphism, User defined Exception handling mechanism
- Illustrating simple inheritance, multi-level inheritance, Exception handling mechanism
- Construct Threads, Event Handling, implement packages, developing applets

Exercise - 1 (Basics)

- Write a JAVA program to display default value of all primitive data type of JAVA
- Write a java program that display the roots of a quadratic equation $ax^2+bx=0$. Calculate the discriminate D and basing on value of D, describe the nature of root.
- Five Bikers Compete in a race such that they drive at a constant speed which may or may not be the same as the other. To qualify the race, the speed of a racer must be more than the average speed of all 5 racers. Take as input the speed of each racer and print back the speed of qualifying racers.

Exercise - 2 (Operations, Expressions, Control-flow, Strings)

- Write a JAVA program to search for an element in a given list of elements using binary search mechanism.
- Write a JAVA program to sort for an element in a given list of elements using bubble sort
- Write a JAVA program to sort for an element in a given list of elements using merge sort.
- Write a JAVA program using StringBuffer to delete, remove character.

Exercise - 3 (Class, Objects)

- a) Write a JAVA program to implement class mechanism. Create a class, methods and invoke them inside main method.
- b) Write a JAVA program to implement constructor.

Exercise - 4 (Methods)

- a) Write a JAVA program to implement constructor overloading.
- b) Write a JAVA program implement method overloading.

Exercise - 5 (Inheritance)

- a) Write a JAVA program to implement Single Inheritance
- b) Write a JAVA program to implement multi level Inheritance
- c) Write a java program for abstract class to find areas of different shapes

Exercise - 6 (Inheritance - Continued)

- a) Write a JAVA program give example for “super” keyword.
- b) Write a JAVA program to implement Interface. What kind of Inheritance can be achieved?

Exercise - 7 (Exception)

- a) Write a JAVA program that describes exception handling mechanism
- b) Write a JAVA program Illustrating Multiple catch clauses

Exercise – 8 (Runtime Polymorphism)

- a) Write a JAVA program that implements Runtime polymorphism
- b) Write a Case study on run time polymorphism, inheritance that implements in above problem

Exercise – 9 (User defined Exception)

- a) Write a JAVA program for creation of Illustrating throw
- b) Write a JAVA program for creation of Illustrating finally
- c) Write a JAVA program for creation of Java Built-in Exceptions
- d) Write a JAVA program for creation of User Defined Exception

Exercise – 10 (Threads)

- a) Write a JAVA program that creates threads by extending Thread class .First thread display “Good Morning “every 1 sec, the second thread displays “Hello “every 2 seconds and the third display “Welcome” every 3 seconds ,(Repeat the same by implementing Runnable)
- b) Write a program illustrating **isAlive** and **join ()**
- c) Write a Program illustrating Daemon Threads.

Exercise - 11 (Threads continuity)

- a) Write a JAVA program Producer Consumer Problem
- b) Write a case study on thread Synchronization after solving the above producer consumer problem

Exercise – 12 (Packages)

- a) Write a JAVA program illustrate class path
- b) Write a case study on including in class path in your os environment of your package.
- c) Write a JAVA program that import and use the defined your package in the previous Problem

Exercise - 13 (Applet)

- a) Write a JAVA program to paint like paint brush in applet.
- b) Write a JAVA program to display analog clock using Applet.
- c) Write a JAVA program to create different shapes and fill colors using Applet.

Exercise - 14 (Event Handling)

- a) Write a JAVA program that display the x and y position of the cursor movement using Mouse.
- b) Write a JAVA program that identifies key-up key-down event user entering text in a Applet.

II Year - II Semester	Skill Oriented Course- II	L	T	P	C
		0	0	4	2
APPLICATIONS OF PYTHON-Pandas					

Course Objectives:

The objective of this lab is to understand the fundamentals of the Pandas library in Python and how it is used to handle data and also develop basic skills in data analysis and visualization

Course Outcomes: By the end of this lab the student is able to

- Use Pandas to create and manipulate data structures like Series and DataFrames.
- Work with arrays, queries, and dataframes
- Query DataFrame structures for cleaning and processing and manipulating files
- Understand best practices for creating basic charts

Perform the following:

- 1) Pandas Installation
2. Creating Data Frames

Exercises: 1

A) Pandas DataSeries:

- 1) Write a Pandas program to create and display a one-dimensional array-like object containing an array of data using Pandas module.
- 2) Write a Pandas program to convert a Panda module Series to Python list and it's type.
- 3) Write a Pandas program to add, subtract, multiple and divide two Pandas Series.
- 4) Write a Pandas program to convert a NumPy array to a Pandas series.

Sample

Series:NumPy

array:

```
[10 20 30 40 50]
```

Converted Pandas

series:0 10

1 20

2 30

3 40

4 50

dtype: int64

B) Pandas DataFrames:

Consider Sample Python dictionary data and list labels:

```
exam_data = {'name': ['Anastasia', 'Dima', 'Katherine', 'James', 'Emily', 'Michael',
'Matthew', 'Laura', 'Kevin', 'Jonas'],
'score': [12.5, 9, 16.5, np.nan, 9, 20, 14.5, np.nan, 8, 19],
'attempts': [1, 3, 2, 3, 2, 3, 1, 1, 2, 1],
'qualify': ['yes', 'no', 'yes', 'no', 'no', 'yes', 'yes', 'no', 'no', 'yes']}
labels = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j']
```

- 1) Write a Pandas program to create and display a DataFrame from a specified dictionary data which has the index labels.

- 2) Write a Pandas program to change the name 'James' to 'Suresh' in name column of the DataFrame.
- 3) Write a Pandas program to insert a new column in existing DataFrame.
- 4) Write a Pandas program to get list from DataFrame column headers.
- 5) Write a Pandas program to get list from DataFrame column headers.

C) Pandas Index:

- 1) Write a Pandas program to display the default index and set a column as an Index in a given dataframe.
- 2) Write a Pandas program to create an index labels by using 64-bit integers, using floating-point numbers in a given dataframe.

D) Pandas String and Regular Expressions:

- 1) Write a Pandas program to convert all the string values to upper, lower cases in a given pandas series. Also find the length of the string values.
- 2) Write a Pandas program to remove whitespaces, left sided whitespaces and right sided whitespaces of the string values of a given pandas series.
- 3) Write a Pandas program to count of occurrence of a specified substring in a DataFrame column.
- 4) Write a Pandas program to swap the cases of a specified character column in a given DataFrame.

E) Pandas Joining and merging DataFrame:

- 1) Write a Pandas program to join the two given dataframes along rows and assign all data.
- 2) Write a Pandas program to append a list of dictionaries or series to a existing DataFrame and display the combined data.
- 3) Write a Pandas program to join the two dataframes with matching records from both sides where available.

F) Pandas Time Series:

- 1) Write a Pandas program to create
 - a) Datetime object for Jan 15 2012.
 - b) Specific date and time of 9:20 pm.
 - c) Local date and time.
 - d) A date without time.
 - e) Current date.
 - f) Time from a datetime.
 - g) Current local time.
- 2) Write a Pandas program to create a date from a given year, month, day and another date from a given string formats.
- 3) Write a Pandas program to create a time-series with two index labels and random values. Also print the type of the index.

G) Pandas Grouping Aggregate:

Consider dataset:

	school	class	name	date_Of_Birth	age	height	weight	address
S1	s001	V	Alberto Franco	15/05/2002	12	173	35	street1
S2	s002	V	Gino Mcneill	17/05/2002	12	192	32	street2
S3	s003	VI	Ryan Parkes	16/02/1999	13	186	33	street3
S4	s001	VI	Eesha Hinton	25/09/1998	13	167	30	street1
S5	s002	V	Gino Mcneill	11/05/2002	14	151	31	street2
S6	s004	VI	David Parkes	15/09/1997	12	159	32	street4

- 1) Write a Pandas program to split the following dataframe into groups based on school code. Also check the type of GroupBy object.
- 2) Write a Pandas program to split the following dataframe by school code and get mean, min, and max value of age for each school.

H) Pandas Styling:

- 1) Create a dataframe of ten rows, four columns with random values. Write a Pandas program to highlight the negative numbers red and positive numbers black.
- 2) Create a dataframe of ten rows, four columns with random values. Write a Pandas program to highlight the maximum value in each column.
- 3) Create a dataframe of ten rows, four columns with random values. Write a Pandas program to highlight dataframe's specific columns.

I) Excel:

- 1) Write a Pandas program to import excel data into a Pandas dataframe.
- 2) Write a Pandas program to find the sum, mean, max, min value of a column of file.

J) Plotting:

- 1) Write a Pandas program to create a horizontal stacked bar plot of opening, closing stock prices of any stock dataset between two specific dates.
- 2) Write a Pandas program to create a histograms plot of opening, closing, high, low stock prices of stock dataset between two specific dates.
- 3) Write a Pandas program to create a stacked histograms plot of opening, closing, high, low stock prices of stock dataset between two specific dates with more bins.

K) Pandas SQL Query:

- 1) Write a Pandas program to display all the records of a student file.
- 2) Write a Pandas program to select distinct department id from employees file.

III Year I Semester		L	T	P	C
		3	0	0	3
COMPUTER NETWORKS					

Course Objectives:

- To provide insight about networks, topologies, and the key concepts.
- To gain comprehensive knowledge about the layered communication architectures (OSI and TCP/IP) and its functionalities.
- To understand the principles, key protocols, design issues, and significance of each layer in ISO and TCP/IP.
- To know the basic concepts of network services and various network applications.

Course Outcomes:

By the end of the course, the student will be able to

- Demonstrate different network models for networking links OSI, TCP/IP, B-ISDN, N-BISDN and get knowledge about various communication techniques, methods and protocol standards.
- Discuss different transmission media and different switching networks.
- Analyze data link layer services, functions and protocols like HDLC and PPP.
- Compare and Classify medium access control protocols like ALOHA, CSMA, CSMA/CD, CSMA/CA, Polling, Token passing, FDMA, TDMA, CDMA protocols
- Determine application layer services and client server protocols working with the client server paradigms like WWW, HTTP, FTP, e-mail and SNMP etc.

UNIT I:

Introduction: Network Types, LAN, MAN, WAN, Network Topologies Reference models- The OSI Reference Model- the TCP/IP Reference Model - A Comparison of the OSI and TCP/IP Reference Models, OSI Vs TCP/IP, Lack of OSI models success, Internet History.

Physical Layer –Introduction to Guided Media- Twisted-pair cable, Coaxial cable and Fiber optic cable and unguided media: Wireless-Radio waves, microwaves, infrared.

UNIT II:

Data link layer: Design issues, **Framing:** fixed size framing, variable size framing, flow control, error control, error detection and correction codes, CRC, Checksum: idea, one's complement internet checksum, services provided to Network Layer, **Elementary Data Link Layer protocols:** simplex protocol, Simplex stop and wait, Simplex protocol for Noisy Channel.

Sliding window protocol: One bit, Go back N, Selective repeat-Stop and wait protocol, Data link layer in HDLC: configuration and transfer modes, frames, control field, point to point protocol (PPP): framing transition phase, multiplexing, multi link PPP.

UNIT - III:

Media Access Control: Random Access: ALOHA, Carrier sense multiple access (CSMA), CSMA with Collision Detection, CSMA with Collision Avoidance, **Controlled Access:** Reservation, Polling, Token Passing, **Channelization:** frequency division multiple Access(FDMA), time division multiple access(TDMA), code division multiple access(CDMA).

Wired LANs: Ethernet, Ethernet Protocol, Standard Ethernet, Fast Ethernet(100 Mbps), Gigabit Ethernet, 10 Gigabit Ethernet.

UNIT - IV:

The Network Layer Design Issues – Store and Forward Packet Switching-Services Provided to the Transport layer- Implementation of Connectionless Service-Implementation of Connection Oriented Service- Comparison of Virtual Circuit

and Datagram Networks, Routing Algorithms-The Optimality principle-Shortest path, Flooding, Distance vector, Link state, TrafficControl Algorithm-Leaky bucket & Token bucket.

Internet Working: How networks differ- How networks can be connected- Tunnelling, internetwork routing, Fragmentation, network layer in the internet – IP protocols-IP Version 4 protocol-IPV4 Header Format, IP addresses .

UNIT –V:

The Transport Layer: Transport layer protocols: Introduction-services- port number-User data gram protocol-User datagram-UDP services-UDP applications-Transmission control protocol: TCP services- TCP features- Segment- A TCP connection- windows in TCP- flow control-Error control, Congestion control in TCP.

Application Layer – World Wide Web: HTTP, Electronic mail-Architecture- web based mail- email security- TELENET- local versus remote Logging-Domain Name System: Name Space, DNS in Internet , - Resolution-Caching- Resource Records- DNS messages- Registrars-security of DNS Name Servers, SNMP.

Text Books:

1. Computer Networks — Andrew S Tanenbaum, Fifth Edition. Pearson Education/PHI
2. Data Communications and Networks – Behrouz A. Forouzan, Fifth Edition TMH.

References Books:

1. Data Communications and Networks- Achut S Godbole, Atul Kahate
2. Computer Networks, Mayank Dave, CENGAGE

III Year I Semester		L	T	P	C
		3	0	0	3
DESIGN AND ANALYSIS OF ALGORITHMS					

Course Objectives:

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

- Ability to understand, analyze and denote time complexities of algorithms
- To introduce the different algorithmic approaches for problem solving through numerous example problems
- Describe the dynamic-programming paradigm and explain when an algorithmic design situation calls for it. Recite algorithms that employ this paradigm. Synthesize dynamic-programming algorithms, and analyze them.
- To provide some theoretical grounding in terms of finding the lower bounds of algorithms and the NP-completeness

Course Outcomes: After the completion of the course, student will be able to

- Analyze the performance of a given algorithm, denote its time complexity using the asymptotic notation for recursive and non-recursive algorithms
- List and describe various algorithmic approaches and Solve problems using divide and conquer & greedy Method
- Synthesize efficient algorithms dynamic programming approaches to solve in common engineering design situations.
- Organize important algorithmic design paradigms and methods of analysis: backtracking, branch and bound algorithmic approaches
- Demonstrate NP- Completeness theory, lower bound theory and String Matching

UNIT I:

Introduction: Algorithm Definition, Algorithm Specification, performance Analysis, Performance measurement, asymptotic notation, Randomized Algorithms.

UNIT II:

Divide and Conquer: General Method, Defective chessboard, Binary Search, finding the maximum and minimum, Merge sort, Quick sort.

The Greedy Method: The general Method, knapsack problem, minimum-cost spanning Trees, Optimal Merge Patterns, Single Source Shortest Paths.

UNIT III:

Dynamic Programming: The general method, multistage graphs, All pairs-shortest paths, optimal Binary search trees, 0/1 knapsack, The traveling salesperson problem.

UNIT IV:

Backtracking: The General Method, The 8-Queens problem, sum of subsets, Graph coloring, Hamiltonian cycles, knapsack problem.

Branch and Bound: General method, applications – Travelling sales person problem, 0/1 knapsack problem- LC Branch and Bound solution, FIFO Branch and Bound solution.

UNIT V:

NP-Hard and NP-Complete problems: Basic concepts, non-deterministic algorithms, NP - Hard and NP-Complete classes, Cook's theorem.

Text Books:

1. Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran, "Fundamentals of Computer Algorithms",

2 Edition, Universities Press.

2. Introduction to Algorithms Thomas H. Cormen, PHI Learning

3. Harsh Bhasin, "Algorithms Design & Analysis", Oxford University Press.

Reference Books:

1. Horowitz E. Sahani S: "Fundamentals of Computer Algorithms", 2nd Publications, 2008.

2. S. Sridhar, "Design and Analysis of Algorithms", Oxford University Press. Edition, Galgotia

III Year I Semester		L	T	P	C
		3	0	0	3
DATA WAREHOUSING AND DATA MINING					

COURSE OUTCOMES:

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

- CO1.** Demonstrate Concepts of knowledge in data warehousing and data mining.
- CO2.** Analyze using data mining techniques to find useful and potential Knowledge.
- CO3.** Design Data Warehouse for OLAP applications for deployment.
- CO4.** Evaluate the usage of association mining techniques on complex data objects.
- CO5.** Select appropriate techniques to measure the interesting patterns from heterogeneous databases.

UNIT I: DATA WAREHOUSING AND ONLINE ANALYTICAL PROCESSING

Data Warehouse, Operational Database Systems versus Data Warehouses, A Multi tired Architecture, A Multidimensional Data Model, Stars, Snowflakes and Fact Constellations: Schemas, Role of Concept hierarchies, Measures, OLAP Operations, From online Analytical processing to Multidimensional Data Mining, Indexing OLAP Data.

UNIT II: DATA MINING AND DATA PREPROCESSING

Introduction to Data Mining, kinds of data, kinds of patterns, major issues in Data Mining, Data Pre-processing, Data Cleaning, Data Integration , Data Reduction, Data Transformation and Discretization.

UNIT III: ASSOCIATIONS AND CLASSIFICATION

Basic Concepts , Frequent itemset Mining Methods, pattern evaluation methods- From Association Mining to Correlation Analysis ,Classification, Decision Tree Introduction, Bayesian Classification Methods, Rule Based Classification, Prediction: Linear Regression.

UNIT IV: CLUSTER ANALYSIS

Cluster Analysis: Types of Data in Cluster Analysis, A Categorization of Major Clustering Methods, Partitioning Methods -k-Means and K-Medoids , Hierarchical methods-Agglomerative and divisive method, Density-Based Method-DBSCAN, Grid-Based Method-STING, Outlier Analysis.

UNIT V: DATA MINING TRENDS

Mining Complex Data Types: Mining sequence data, Mining other kinds of data: Spatial, Text, Multimedia and Web data, Data Mining Trends.

TEXT BOOK:

T1. Jiawei Han, Micheline Kamber and Jian Pei, *Data Mining: Concepts and Techniques*, Elsevier, Third Edition, 2013.

REFERENCE BOOKS:

1. K.P. Soman, Shyam Diwakar and V. Ajay, *Insight into Data mining Theory and Practice*, Easter Economy Edition, Prentice Hall of India, 2006.
2. G. K. Gupta, *Introduction to Data Mining with Case Studies*, Easter Economy Edition, Prentice Hall of India, 2006.
3. Tan P.N, Steinbach M.and Kumar V., *Introduction to Data mining*, Addison-Weseley, 2006.

OE - I

III Year – I Semester		L	T	P	C
		3	0	0	3
ARTIFICIAL INTELLIGENCE Professional Elective-1					

Course Objectives

- To provide a strong foundation of fundamental concepts in Artificial Intelligence.
- To provide a basic exposition to the goals and methods of Artificial Intelligence.
- To apply the techniques in applications which involve perception, reasoning and learning.

Course Outcomes:

Upon successful completion of the course, the student will be able to:

- Enumerate the history and foundations of Artificial Intelligence
- Apply the basic principles of AI in problem solving
- Choose the appropriate representation of Knowledge
- Solve the problems with uncertainty using probability
- Examine the Scope of AI and its societal implications

UNIT I

Introduction: What Is AI? The Foundations of Artificial Intelligence, The History of Artificial Intelligence, The State of the Art, Agents and Environments, Good Behavior: The Concept of Rationality, The Nature of Environments, The Structure of Agents.

UNIT II

Problem Solving: Problem-Solving Agents, Example Problems, Searching for Solutions, Uninformed Search Strategies, Informed (Heuristic) Search Strategies, Local Search Algorithms and Optimization Problems, Searching with Nondeterministic Actions.

UNIT III

Knowledge Representation: Knowledge-Based Agents, Logic, Propositional Logic: A Very Simple Logic, Ontological Engineering, Categories and Objects, Events, Mental Events and Mental Objects, Reasoning Systems for Categories, The Internet Shopping World.

UNIT IV

Uncertain Knowledge and Reasoning: Acting under Uncertainty, Basic Probability Notation, Inference Using Full Joint Distributions, Independence, Bayes' Rule and Its Use, Representing Knowledge in an Uncertain Domain, The Semantics of Bayesian Networks.

UNIT V

AI present and Future: Weak AI: Can Machines Act Intelligently? Strong AI: Can Machines Really Think? The Ethics and Risks of Developing Artificial Intelligence, Agent Components, Agent Architectures, Are We Going in the Right Direction? What If AI Does Succeed?

Text Books:

- 1) Stuart Russell and Peter Norvig, "Artificial Intelligence: A Modern Approach" , 3rd Edition, Pearson.
- 2) Elaine Rich and Kevin Knight, "Artificial Intelligence", Tata McGraw Hill

Reference Books:

- 1) Saroj Kaushik, "Artificial Intelligence", Cengage Learning India, 2011
- 2) David Poole and Alan Mackworth, "Artificial Intelligence: Foundations for Computational Agents", Cambridge University Press 2010.
- 3) Trivedi, M.C., "A Classical Approach to Artificial Intelligence", Khanna Publishing House, Delhi.

Web Resources:

- 1) <https://nptel.ac.in/courses/106105077>
- 2) <https://nptel.ac.in/courses/106106126>
- 3) <https://aima.cs.berkeley.edu>
- 4) https://ai.berkeley.edu/project_overview.html

III Year – I Semester		L	T	P	C
		3	0	0	3

Compiler Design

- To study the various phases in the design of a compiler
- To understand the design of top-down and bottom-up parsers
- To understand syntax directed translation schemes
- To introduce LEX and YACC tools
- To learn to develop algorithms to generate code for a target machine

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

- Design, develop, and implement a compiler for any language
- Use LEX and YACC tools for developing a scanner and a parser
- Design and implement LL and LR parsers
- Design algorithms to perform code optimization in order to improve the performance of a program in terms of space and time complexity
- Apply algorithms to generate machine code

UNIT-1

Language Processors, the structure of a compiler, the science of building a compiler, programming language basics.

Lexical Analysis: The Role of the Lexical Analyzer, Input Buffering, Recognition of Tokens, The Lexical-Analyzer Generator Lex, Finite Automata, From Regular Expressions to Automata, Design of a Lexical-Analyzer Generator, Optimization of DFA-Based Pattern Matchers.

UNIT-2

Syntax Analysis: Introduction, Context-Free Grammars, Writing a Grammar, Top-Down Parsing, Recursive and Non recursive top down parsers, Bottom-Up Parsing, Introduction to LR Parsing: Simple LR, More Powerful LR Parsers, Using Ambiguous Grammars, Parser Generators.

UNIT-3

Syntax-Directed Definitions, Evaluation Orders for SDD's, Applications of Syntax-Directed Translation, Syntax-Directed Translation Schemes, and Implementing L-Attributed SDD's.

Intermediate-Code Generation: Variants of Syntax Trees, Three-Address Code, Types and Declarations, Type Checking, Control Flow, Back patching, Switch-Statements, Intermediate Code for Procedures.

UNIT-4

Run-Time Environments: Storage organization, Stack Allocation of Space, Access to Nonlocal Data on the Stack, Heap Management, Introduction to Garbage Collection, Introduction to TraceBased Collection.

Machine-Independent Optimizations: The Principal Sources of Optimization, Introduction to Data-Flow Analysis, Foundations of Data-Flow Analysis, Constant Propagation, PartialRedundancy Elimination, Loops in Flow Graphs.

UNIT-5

Code Generation: Issues in the Design of a Code Generator, The Target Language, Addresses in the Target Code, Basic Blocks and Flow Graphs, Optimization of Basic Blocks, A Simple Code Generator.

Machine-dependent Optimizations: Peephole Optimization, Register Allocation and Assignment, Dynamic Programming Code-Generation.

TEXT BOOKS:

1. Compilers: Principles, Techniques and Tools, Second Edition, Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman, Pearson.
2. Compiler Construction-Principles and Practice, Kenneth C Loudon, Cengage Learning.

REFERENCE BOOKS:

1. Modern compiler implementation in C, Andrew W Appel, Revised edition, Cambridge University Press.
2. The Theory and Practice of Compiler writing, J. P. Tremblay and P. G. Sorenson, TMH
3. Writing compilers and interpreters, R. Mak, 3rd edition, Wiley student edition.

~~Global State Consistency in Distributed Systems~~

III Year I Semester	L	T	P	C
	3	0	0	3
DISTRIBUTED SYSTEMS Professional Elective-1				

Course Objectives:

- To understand the foundations of distributed systems.
- To learn issues related to clock Synchronization and the need for global state in distributed systems
- To learn distributed mutual exclusion and deadlock detection algorithms
- To understand the significance of agreement, fault tolerance and recovery protocols in Distributed Systems
- To learn the characteristics of peer-to-peer and distributed shared memory systems

Course Outcomes:

- At the end of the course, the students will be able to:
- Elucidate the foundations and issues of distributed systems
- Illustrate the various synchronization issues and global state for distributed systems
- Illustrate the Mutual Exclusion and Deadlock detection algorithms in distributed systems
- Describe the agreement protocols and fault tolerance mechanisms in distributed systems
- Describe the features of peer-to-peer and distributed shared memory systems

UNIT I

Distributed Systems: Definition, Relation to computer system components, Motivation, Relation to parallel systems, Message-passing systems versus shared memory systems, Primitives for distributed communication, Synchronous versus asynchronous executions, Design issues and challenges.

A model of distributed computations: A distributed program, A model of distributed executions, Models of communication networks, Global state, Cuts, Past and future cones of an event, Models of process communications.

Logical Time: A framework for a system of logical clocks, Scalar time, Vector time, Physical clock synchronization: NTP.

UNIT II

Message Ordering & Snapshots: Message ordering and group communication: Message ordering paradigms, Asynchronous execution with synchronous communication, Synchronous program order on an asynchronous system, Group communication, Causal order (CO), Total order. Global state and snapshot recording algorithms: Introduction, System model and definitions, Snapshot algorithms for FIFO channels.

UNIT III

Distributed Mutex & Deadlock: Distributed mutual exclusion algorithms: Introduction – Preliminaries – Lamport’s algorithm – Ricart-Agrawala algorithm – Maekawa’s algorithm – Suzuki–Kasami’s broadcast algorithm. Deadlock detection in distributed systems: Introduction – System model – Preliminaries – Models of deadlocks – Knapp’s classification – Algorithms for the single resource model, the AND model and the OR model.

UNIT IV

Recovery & Consensus: Check pointing and rollback recovery: Introduction – Background and definitions – Issues in failure recovery – Checkpoint-based recovery – Log-based rollback recovery – Coordinated check pointing algorithm – Algorithm for asynchronous check pointing and recovery. Consensus and agreement algorithms: Problem definition – Overview of results – Agreement in a failure – free system – Agreement in synchronous systems with failures.

UNIT V

Peer-to-peer computing and overlay graphs: Introduction – Data indexing and overlays – Chord – Content addressable networks – Tapestry.

Distributed shared memory: Abstraction and advantages – Memory consistency models – Shared memory Mutual Exclusion.

Text Books:

1) Distributed Systems Concepts and Design, George Coulouris, Jean Dollimore and Tim Kindberg, Fifth Edition, Pearson Education, 2012.

2) Distributed computing: Principles, algorithms, and systems, Ajay D Kshemkalyani and Mukesh Singhal, Cambridge University Press, 2011.

Reference Books:

1) Distributed Operating Systems: Concepts and Design, Pradeep K Sinha, Prentice Hall of

India, 2007.

2) Advanced concepts in operating systems. Mukesh Singhal and Niranjana G. Shivaratri,

McGraw-Hill, 1994.

3) Distributed Systems: Principles and Paradigms, Tanenbaum A.S., Van Steen M., Pearson

Education, 2007.

III Year I Semester		L	T	P	C
				3	1.5
DATA WAREHOUSING AND DATA MINING LAB					

Lab Experiments:

1. Create a dataset using ARFF and CSV formats and load into the Weka Explorer.
2. Perform the following preprocessing filters on 'Weather' dataset. (i) Add (ii) Remove (iii) Discretize (iv) Replace Missing values (v) Normalize
3. List all the categorical attributes and the real-valued attributes separately in 'German credit' data set.
4. Generate strong Association rules by using Apriori algorithm on 'German_credit' dataset with Min_Sup=60% and Min_Conf=80%.
5. (i) Implement the Classification using Decision Tree algorithm on 'Weather' dataset. Draw the confusion matrix and report the model with accuracy. (ii) Implement Bayesian Classification and analyze the results on 'iris' Dataset.
6. (i) Implement Simple Linear Regression on an 'Employee' dataset. (ii) Demonstrate the simple k-Means clustering algorithm on 'iris' dataset.
7. (i) Rank the performance of j48, PART and oneR Algorithms on 'Weather' dataset using Experimenter. (ii) Build a Knowledge Flow model using Weka tool. II. Experiments using Rapid Miner
8. (i) Implement the Classification using Decision Tree algorithm on 'Weather' dataset. Draw the confusion matrix and report the model with accuracy. (ii) Implement Bayesian Classification and analyze the results on 'iris' Dataset. the values {6, 9, 5, 2, 8, 15, 24, 14, 7, 8, 5, 2} in the given order. b) Display the nodes of the tree using inorder, preorder and postorder traversal techniques. c) Display the smallest number stored in the tree. d) Search the tree for a given number.
9. (i) Implement Simple Linear Regression on an 'Employee' dataset. (ii) Demonstrate the simple k-Means clustering algorithm on 'iris' dataset.
10. (i) Rank the performance of j48, PART and oneR Algorithms on 'Weather' dataset using Experimenter. (ii) Build a Knowledge Flow model using Rapid Miner tool.
11. Mini Project

REFERENCE BOOKS:

1. Ian. H. Witten and Eibe Frank, Data Mining: Practical Machine Learning Tools and Techniques, Elsevier Publication, 2nd Edition, 2005.
2. Pang-Ning Tan, Vipin Kumar and Michael Steinbach, Introduction to Data Mining, Pearson Education, 2006.
3. Jiawei Han and Micheline Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann Publishers, 2nd Edition, 2006.

III Year I Semester		L	T	P	C
				3	1.5
COMPUTER NETWORKS LAB					

Course Objectives:

- Learn basic concepts of computer networking and acquire practical notions of protocols with the emphasis on TCP/IP. A lab provides a practical approach to Ethernet/Internet networking: networks are assembled, and experiments are made to understand the layered architecture and how do some important protocols work

Course Outcomes:

By the end of the course student will be able to

- Know how reliable data communication is achieved through data link layer.
- Suggest appropriate routing algorithm for the network.
- Provide internet connection to the system and its installation.
- Work on various network management tools

List of Experiments:

1. Study of Network devices in detail and connect the computers in Local Area Network.
2. Write a Program to implement the data link layer framing methods such as
 - i) Character stuffing
 - ii) bit stuffing.
3. Write a Program to implement data link layer framing method checksum.
4. Write a program for Hamming Code generation for error detection and correction.
5. Write a Program to implement on a data set of characters the three CRC polynomials – CRC 12, CRC 16 and CRC CCIP.
6. Write a Program to implement Sliding window protocol for Goback N.
7. Write a Program to implement Sliding window protocol for Selective repeat.
8. Write a Program to implement Stop and Wait Protocol.
9. Write a program for congestion control using leaky bucket algorithm
10. Write a Program to implement Dijkstra's algorithm to compute the Shortest path through a graph.
11. Write a Program to implement Distance vector routing algorithm by obtaining routing table at each node (Take an example subnet graph with weights indicating delay between nodes).
12. Write a Program to implement Broadcast tree by taking subnet of hosts.
13. Wireshark
 - i. Packet Capture Using Wire shark
 - ii. Starting Wire shark
 - iii. Viewing Captured Traffic
 - iv. Analysis and Statistics & Filters.

III Year I Semester		L	T	P	C
		3			3
Dev Ops Skill Oriented Course – III					

Course Objectives:

- DevOps improves collaboration and productivity by automating infrastructure and workflows and continuously measuring applications performance
- Course Outcomes:

At the end of the course, student will be able to

- Enumerate the principles of continuous development and deployment, automation of configuration management, inter-team collaboration, and IT service agility
- Describe DevOps & DevSecOps methodologies and their key concepts
- Illustrate the types of version control systems, continuous integration tools, continuous monitoring tools, and cloud models
- Set up complete private infrastructure using version control systems and CI/CD tools

UNIT I

Phases of Software Development life cycle. Values and principles of agile software development.

UNIT II

Fundamentals of DevOps: Architecture, Deployments, Orchestration, Need, Instance of applications, DevOps delivery pipeline, DevOps eco system.

UNIT III

DevOps adoption in projects: Technology aspects, Agiling capabilities, Tool stack implementation, People aspect, processes

UNIT IV

CI/CD: Introduction to Continuous Integration, Continuous Delivery and Deployment , Benefits of CI/CD, Metrics to track CICD practices

UNIT V

Devops Maturity Model: Key factors of DevOps maturity model, stages of Devops maturity model, DevOps maturity Assessment

Text Books:

- 1) The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in

Technology Organizations, Gene Kim , John Willis , Patrick Debois , Jez Humb,1st Edition, O'Reilly publications, 2016.

2) What is Devops? Infrastructure as code, 1st Edition, Mike Loukides ,O'Reilly publications, 2012.

Reference Books:

1) Building a DevOps Culture, 1st Edition, Mandi Walls, O'Reilly publications, 2013.

2) The DevOps 2.0 Toolkit: Automating the Continuous Deployment Pipeline With

Containerized Microservices, 1st Edition, Viktor Farcic, CreateSpace Independent Publishing Platform publications, 2016

3) Continuous Delivery: Reliable Software Releases Through Build, Test, and Deployment Automation, 1st Edition, Jez Humble and David Farley, 2010.

4) Achieving DevOps: A Novel About Delivering the Best of Agile, DevOps, and microservices, 1st Edition, Dave Harrison, Knox Lively, Apress publications, 2019

e-Resources:

1) <https://www.javatpoint.com/devops>

2) <https://github.com/nkatre/Free-DevOps-Books-1/blob>

Employability Skills-I

Summer Internship 2 Months

(Mandatory) after second year (to beevaluated during V semester

III Year I I Semester		L	T	P	C
		3	0	0	3
BIG DATA ANALYTICS					

Course Objective:

- The main objective of this course is to provide the student with the basic understanding of Data Analysis.
- Program the related analysis of different types of data and using Hadoop programming to predict the trends.

Course Outcomes:

- Understand the concepts of Big Data Analytics.
- Apply machine learning algorithms for Big Data Analytics
- Apply text categorization algorithms
- Solve the Big Data Analytics problems using various technologies and tools
- Apply Sqoop, and Zookeeper tools for solving the Data Analysis problems.

UNIT-I: Introduction to Big Data Analytics: Big Data Overview, State of the Practice in Analytics, Key Roles for the New Big Data Ecosystem, Examples of Big Data Analytics. Data Analytics Lifecycle: Data Analytics Lifecycle Overview, Discovery, Data Preparation, Model Planning, Model Building, Communicate Results, Operationalize.

UNIT-II: Introduction to Hadoop: Data storage and analysis, Comparison with other systems, A Brief History of Hadoop, Apache Hadoop and the Hadoop Ecosystem. **Hadoop Distributed File system:** The Design of HDFS, HDFS concepts, Command-Line Interface, Hadoop file systems, The Java Interface, Data Flow, Hadoop Archives.

UNIT-III: Hadoop I/O: Data Integrity in HDFS, Local-FileSystem, Checksum-FileSystem, Compression, Serialization, File-Based Data Structures, Sequential File, Map File. **Map-Reduce:** A Weather Dataset, Analyzing the data with Hadoop, Map and Reduce, Java Map and Reduce, Scaling Out, Hadoop Streaming, Hadoop Pipes. **Map-Reduce Types and Formats:** Map-Reduce Types, Input Formats, Output Formats.

UNIT-IV: Pig: Pig-Installation and Running of Pig, Execution Types, Running Pig Programs, Pig Latin Editors, Comparison with databases, Pig Latin, Functions, Data Processing Operators. **Hive:** Installing Hive, An Example, Running Hive, Comparison with Traditional Databases, Hive-QL, Tables, Querying Data.

UNIT-V: Zookeeper: Zookeeper Service, Building applications with Zookeeper, Zookeeper in production. **Sqoop:** Database Imports, working with Imported Data, Importing Large Objects, Performing an Export.

Text Books:

1. Data Science and Big Data Analytics, EMC2 Education Services, wiley,2015.
2. Tom White,Hadoop: The Definitive Guide, 4rd Edition, O'reilly Publication, 2015.
3. John A. Rice, Mathematical Statistics and Data Analysis, 3rd Edition, Cengage.

Reference Books:

- 1.Franks, Taming, The Big Data Tidal Wave, 1st Edition, Wiley, 2012.
2. VigneshPrajapati, Big Data Analytics with R and Hadoop, packet publishing 2013Bill.
3. Frank J. Ohlhorst, Big Data Analytics

III Year I I Semester		L	T	P	C
		3	0	0	3
CLOUD COMPUTING					

COURSE OBJECTIVES:

- To impart knowledge on Virtualization and Cloud computing concepts.
- To understand the basic knowledge on Cloud Models and Services.
- To develop cloud environments suitable for societal services through Cloud platforms.

COURSE OUTCOMES: After successful completion of the course, students will be able to:

- CO1. Demonstrate basic concepts and terminologies of Cloud Computing and Virtualization.
- CO2. Understand Cloud deployment models, Service models and Architectures.
- CO3. Analyze Cloud services, Applications and Capacity Planning.
- CO4. Use different PaaS application frameworks to construct Cloud applications.
- CO5. Develop web applications through Google, Microsoft and Amazon web services as per societal needs.

UNIT I– FUNDAMENTAL CLOUD COMPUTING AND VIRTUALIZATION
Cloud Computing: Origin and influences, Basic concepts and terminology, Goals and benefits, Risks and challenges, Roles and boundaries and Cloud characteristics.

Introduction to Virtualization: Characteristics, Taxonomy of virtualization technologies, Pros and cons of virtualization, Virtualization Technologies: Xen, VMware and Hyper-V.

UNIT II– UNDERSTANDING CLOUD MODELS AND ARCHITECTURES **Cloud Models:** NIST model, Cloud Cube model, Deployment models: Public, Private, Hybrid and Community; Service models: IaaS, PaaS and SaaS. **Understanding Cloud Architecture:** Exploring the Cloud Computing Stack: Composability, Infrastructure, Platforms, Virtual Appliances, Communication Protocols, Applications; Connecting to the Cloud: The Jolicloud Netbook OS and Chromium OS - The Browser as an Ope

rating System.

UNIT III – UNDERSTANDING CLOUD SERVICES, APPLICATIONS AND CAPACITY PLANNING

Understanding Cloud Services and Applications Infrastructure as a Service (IaaS): IaaS workloads, Pods, aggregation, and silos; Platform as a Service (PaaS), Software as a Service (SaaS): SaaS characteristics, Open SaaS and SOA, Salesforce.com and CRM SaaS; Identity as a Service (IDaaS): Identity, Networked identity service classes, Identity system codes of conduct, IDaaS interoperability; Compliance as a Service (CaaS).

Capacity Planning: Defining Baseline and Metrics: Baseline measurements, System metrics, Load Testing, Resource ceilings, Server and instance types; Network Capacity and Scaling.

UNIT IV – EXPLORING PLATFORM AS A SERVICE (PaaS)

PaaS Application Frameworks: Drupal, Eccentex AppBase 3.0, LongJump, Squarespace, WaveMaker and Wolf Frameworks.

Exploring Platform as a Service using Google Web Services: Surveying the Google Application Portfolio, Google Toolkit and Working with the Google App Engine.

Exploring Platform as a Service using Microsoft Cloud Services: Exploring Microsoft Cloud Services, Defining the Windows Azure Platform, Windows Live: Windows Live Essentials, Windows Live Home and Windows Live for Mobile.

UNIT V – EXPLORING INFRASTRUCTURE AS A SERVICE (IaaS)

Understanding Amazon Web Services, Amazon Web Service Components and Services, Working with the Elastic Compute Cloud (EC2): Amazon Machine Images, Pricing models, System images and software, Creating an account and instance on EC2; Working with Amazon Storage Systems: Amazon Simple Storage System (S3), Amazon Elastic Block Store (EBS) and CloudFront; Understanding Amazon Database Services: Amazon SimpleDB, Amazon Relational Database Service (RDS) and Choosing a database for AWS.

TEXT BOOKS:

1. Barrie Sosinsky, “*Cloud Computing Bible*”, Wiley India Pvt Ltd, 2011 (Reprint 2017).
2. Thomas Erl and RicardoPuttini, “*Cloud Computing- Concepts, Technology and Architecture*”, Pearson, 2014 (Seventh Impression 2017).

REFERENCE BOOKS:

1. Rajkumar Buyya, Christian Vecchiloa and S Thamarai Selvi, “*Mastering Cloud Computing*”, McGraw Hill Education, 2013 (Reprint 2017).
2. George Reese, “*Cloud Application and Architectures*”, O’Reilly, 2009 (Reprint 2017).

ADDITIONAL LEARNING RESOURCES:

1. “Exploring the Google Toolkit”, <https://code.google.com/>, drafted on 23 December, 2019.

2. "Understanding Amazon Web Services", <https://aws.amazon.com/>, drafted on 23 December, 2019.
3. "Exploring Microsoft Cloud Services", <https://www.microsoft.com/windowsazure>, drafted on 23 December, 2019.

III Year II Semester				
	L	T	P	C
CRYPTOGRAPHY AND NETWORK SECURITY				
3 0 0 3				

Course Objectives:

The main objectives of this course are to explore the working principles and utilities of various cryptographic algorithms including secret key cryptography, hashes and message digests, public key algorithms, design issues and working principles of various authentication protocols and various secure communication standards including Kerberos, IPsec, and SSL/TLS.

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

- Explain different security threats and countermeasures and foundation course of cryptography mathematics.
- Classify the basic principles of symmetric key algorithms and operations of some symmetric key algorithms and asymmetric key cryptography
- Revise the basic principles of Public key algorithms and Working operations of some Asymmetric key algorithms such as RSA, ECC and some more
- Design applications of hash algorithms, digital signatures and key management techniques
- Determine the knowledge of Application layer, Transport layer and Network layer security Protocols such as PGP, S/MIME, SSL,TSL, and IPsec .

UNIT I:

Basic Principles : Security Goals, Cryptographic Attacks, Services and Mechanisms, Mathematics of Cryptography. **Security Policies Foundations of modern cryptography**

UNIT II:

Symmetric Encryption: Mathematics of Symmetric Key Cryptography, Introduction to Modern Symmetric Key Ciphers, Data Encryption Standard, Advanced Encryption Standard.

UNIT III:

Asymmetric Encryption: Mathematics of Asymmetric Key Cryptography, Asymmetric Key Cryptography, **Fermat's and Euler's Theorem , Chinese Remainder Theorem**

UNIT IV:

Data Integrity, Digital Signature Schemes & Key Management: Message Integrity and Message Authentication, Cryptographic Hash Functions, Digital Signature, Key Management.

UNIT V:

Network Security-I: Security at application layer: PGP and S/MIME, Security at the Transport Layer: SSL and TLS, **Network Security-II :** Security at the Network Layer: IPSec, System Security

Text Books:

1. Cryptography and Network Security, 3rd Edition Behrouz A Forouzan, Deb deep Mukhopadhyay, McGraw Hill,2015
2. Cryptography and Network Security,4th Edition, William Stallings, (6e) Pearson,2006
3. Everyday Cryptography, 1st Edition, Keith M.Martin, Oxford,2016

Reference Books:

1. Network Security and Cryptography, 1st Edition, Bernard Meneges, Cengage Learning,2018

III Year – II Semester	L	T	P	C
	3	0	0	3
MOBILE COMPUTING (Professional Elective-II)				

Course Objectives:

- To understand the fundamentals of mobile communication
- To understand the architecture of various Wireless Communication Networks
- To understand the significance of different layers in mobile system Course Contents

Course Outcomes: Upon completion of the course, the students will be able to:

- Develop a strong grounding in the fundamentals of mobile Networks
- Apply knowledge in MAC, Network, and Transport Layer protocols of Wireless Network
- Comprehend, design, and develop a lightweight network stack
- Analyze the Mobile Network Layer system working
- Explain about the WAP Model

UNIT I: Introduction to Wireless Networks: Applications, History, Simplified Reference Model, Wireless transmission, Frequencies, Signals, Antennas, Signal propagation, Multiplexing, Modulation, Spread spectrum, Cellular Systems: Frequency Management and Channel Assignment, types of hand-off and their characteristics.

UNIT II: MAC – Motivation, SDMA, FDMA, TDMA, CDMA, Telecommunication Systems, GSM: Architecture Location tracking and call setup, Mobility management, Handover, Security, GSM, SMS, International roaming for GSM, call recording functions, subscriber and service data management, DECT, TETRA, UMTS, IMT-2000.

UNIT III: Wireless LAN: Infrared vs. Radio transmission, Infrastructure, Adhoc Network, IEEE 802.11WLAN Standards, Architecture, Services, HIPERLAN, Bluetooth Architecture & protocols.

UNIT IV: Mobile Network Layer: Mobile IP, Dynamic Host Configuration Protocol, Mobile Transport Layer, Traditional TCP, Indirect TCP, Snooping TCP, Mobile TCP, Fast retransmit/Fast recovery, Transmission/Time-out freezing, Selective retransmission, Transaction Oriented TCP.

UNIT V: Support for Mobility: Wireless Application Protocol: Architecture, Wireless Datagram Protocol, Wireless Transport Layer Security, Wireless Transaction Protocol, Wireless Session Protocol, Wireless Application Environment, Wireless Markup Language, WML Scripts, Wireless Telephone Application.

Text Books:

1. Jochen Schiller, "Mobile Communication", Second Edition, Pearson Education, 2008.

Reference Books:

1. William Stallings, "Wireless Communications and Networks", Second Edition, Pearson Education, 2004.
2. C. Siva Ram Murthy, B. S. Manoj, "Adhoc Wireless Networks: Architectures and Protocols", Second Edition, Pearson Education, 2008.

III Year –II Semester		L	T	P	C
		3	0	0	3
SOFTWARE PROJECT MANAGEMENT (Professional Elective-II)					

Course Objectives:

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

- To describe and determine the purpose and importance of project management from the perspectives of planning, tracking and completion of project
- To compare and differentiate organization structures and project structures
- To implement a project to manage project schedule, expenses and resources with the application of suitable project management tools

Course Outcomes:

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

- Apply the process to be followed in the software development life-cycle models.
- Apply the concepts of project management & planning.
- Implement the project plans through managing people, communications and change
- Conduct activities necessary to successfully complete and close the Software projects
- Implement communication, modeling, and construction & deployment practices in software development.

UNIT I

Conventional Software Management: The waterfall model, conventional software Management performance.

Evolution of Software Economics: Software Economics, pragmatic software cost estimation.

Improving Software Economics: Reducing Software product size, improving software processes, improving team effectiveness, improving automation, Achieving required quality, peer inspections.

UNIT II

The Old Way and The New: The principles of conventional software Engineering, principles of modern software management, transitioning to an iterative process.

Life Cycle Phases: Engineering and production stages, inception, Elaboration, construction, transition phases.

Artifacts of The Process: The artifact sets, Management artifacts, Engineering artifacts, programmatic artifacts.

UNIT III

Model Based Software Architectures: A Management perspective and technical perspective.

Work Flows of the Process: Software process workflows, Iteration workflows. Checkpoints of the Process: Major mile stones, Minor Milestones, Periodic status assessments.

UNIT IV

Iterative Process Planning: Work breakdown structures, planning guidelines, cost and schedule estimating, Iteration planning process, Pragmatic planning.

Project Organizations and Responsibilities: Line-of-Business Organizations, Project Organizations, evolution of Organizations.

UNIT V

Process Automation: Automation Building blocks, The Project Environment.

Project Control and Process Instrumentation: The seven core Metrics, Management indicators, quality indicators, life cycle expectations, pragmatic Software Metrics, Metrics automation.

Project Estimation and Management: COCOMO model, Critical Path Analysis, PERT technique, Monte Carlo approach (Text book 2)

Text Books:

- 1) Software Project Management, Walker Royce, Pearson Education, 2005.
- 2) Software Project Management, Bob Hughes, 4th edition, Mike Cotterell, TMH.

Reference Books:

- 1) Software Project Management, Joel Henry, Pearson Education.
 - 2) Software Project Management in practice, Pankaj Jalote, Pearson Education, 2005.
- Effective Software Project Management, Robert K. Wysocki, Wiley, 2006.

III Year – II Semester		L	T	P	C
		3	0	0	3
OBJECT ORIENTED ANALYSIS AND DESIGN (Professional Elective II)					

Course Objectives: The main objective is the students to

- Become familiar with all phases of OOAD.
- Master the main features of the UML.
- Master the main concepts of Object Technologies and how to apply them at work and develop the ability to analyze and solve challenging problem in various domains.
- Learn the Object design Principles and understand how to apply them towards Implementation.

Course Outcomes: After finishing this course student will be able to:

- Analyze the nature of complex system and its solutions.
- Illustrate & relate the conceptual model of the UML, identify & design the classes and relationships
- Analyze & Design Class and Object Diagrams that represent Static Aspects of a Software System and apply basic and Advanced Structural Modeling Concepts for designing real time applications.
- Analyze & Design behavioral aspects of a Software System using Use Case, Interaction and Activity Diagrams.
- Analyze & Apply techniques of State Chart Diagrams and Implementation Diagrams to model behavioral aspects and Runtime environment of Software Systems.

UNIT I:

Introduction: The Structure of Complex systems, The Inherent Complexity of Software, Attributes of Complex System, Organized and Disorganized Complexity, Bringing Order to Chaos, Designing Complex Systems. **Case Study:** System Architecture: Satellite-Based Navigation

UNIT II:

Introduction to UML: Importance of modeling, principles of modeling, object oriented modeling, conceptual model of the UML, Architecture, and Software Development Life Cycle. **Basic Structural Modeling:** Classes, Relationships, common Mechanisms, and diagrams. **Case Study:** Control System: Traffic Management.

UNIT III:

Class & Object Diagrams: Terms, concepts, modeling techniques for Class & Object Diagrams. **Advanced Structural Modeling:** Advanced classes, advanced relationships, Interfaces, Types and Roles, Packages. **Case Study:** AI: Cryptanalysis.

UNIT IV:

Basic Behavioral Modeling-I: Interactions, Interaction diagrams Use cases, Use case Diagrams, Activity Diagrams. **Case Study:** Web Application: Vacation Tracking System

UNIT V:

Advanced Behavioral Modeling: Events and signals, state machines, processes and Threads, time and space, state chart diagrams. **Architectural Modeling:** Component, Deployment, Component diagrams and Deployment diagrams

Case Study: Weather Forecasting

Text Books:

1. Grady BOOCH, Robert A. Maksimchuk, Michael W. ENGLE, Bobbi J. Young, Jim Conallen, Kellia Houston , "Object- Oriented Analysis and Design with Applications", 3rd edition,2013, PEARSON.
2. Grady Booch, James Rumbaugh, Ivar Jacobson: The Unified Modeling Language User Guide,Pearson Education.

Reference Books:

1. Meilir Page-Jones: Fundamentals of Object Oriented Design in UML, Pearson Education.
2. Pascal Roques: Modeling Software Systems Using UML2, WILEY- Dreamtech India Pvt. Ltd.
3. Atul Kahate: Object Oriented Analysis & Design, The McGraw-Hill Companies.

Appling UML and Patterns: An introduction to Object – Oriented Analysis and Design andUnified Process, Craig Larman, Pearson Education.

III Year I Semester		L	T	P	C
		3	0	0	3
INTERNET OF THINGS Open Elective-II					

Course Objective:

- The main objective of this course is to provide the student with the basic understanding of Embedded software
- Program the related analysis of different types of smart devices can be built by using Arduino and Rasberry-Pi boards.

Course Outcomes:

- Summarize the genesis and impact of IoT applications, architectures in real world.
- Apply diverse methods in deploying smart objects and connecting them to network.
- Construct simple applications using Arduino.
- Analyze different protocols and select which protocol can be used for a specific application.
- Identify and develop a solution for a given application using APIs

UNIT-I: Genesis of IoT, IoT and Digitization, IoT Impact-Connected roadways, Smart connected buildings, Convergence of IT and IoT, IoT Challenges, Comparing IoT Architectures - OneM2M IoT Architecture and IoTWF Architecture, A Simplified IoT Architecture.

UNIT-II: Smart Objects: The Things in IoT- Sensors, Actuators, and Smart Objects, Sensor Networks- Advantages and Disadvantages, Communications Criteria-Range, Frequency bands, Power consumption, Topology, IoT Access Technologies- IEEE 802.15.4, IEEE 1901.2a, IEEE 802.11ah (only Standardization and Alliances, Physical Layer, MAC Layer and Topology)

UNIT-III: Embedded Computing Basics- Microcontrollers, System-on-Chips, Choosing Your Platform, Arduino- Developing on the Arduino, Some Notes on the Hardware, Openness

UNIT-IV: Communication in the IoT: Internet Principles, Internet Communications: An Overview- IP, TCP, The IP Protocol Suite (TCP/IP), UDP, IP Addresses- DNS, Static IP Address Assignment, Dynamic IP Address Assignment, IPv6, MAC Addresses, TCP and UDP Ports- An Example: HTTP Ports, Other Common Ports.

UNIT-V: Application Layer Protocols- HTTP, HTTPS: Encrypted HTTP, Other Application Layer Protocols. Prototyping Online Components: Getting Started with an API, Writing a New API, Real-Time Reactions, other Protocols.

Text Books:

4. Adrian McEwen, Hakim Cassimally - Designing the Internet of Thing Wiley Publications, 2012.
5. David Hanes, Gonzalo Salgueiro, Patrick Grossetete, Robert Barton, Jerome Henry, IIoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things, 1st Edition, Pearson Education (Cisco Press Indian Reprint). (ISBN: 978-9386873743)

Reference Books:

1. Arshdeep Bahga, Vijay Madisetti - Internet of Things: A Hands-On Approach, Universities Press, 2014
2. Srinivasa K G, Internet of Things, CENGAGE Learning India, 2017

III Year I I Semester	L	T	P	C
	3	0	0	3
BIG DATA ANALYTICS LAB				

Course Outcomes:

- Understand the concepts of Big Data Analytics.
- Apply machine learning algorithms for Big Data Analytics
- Apply text categorization algorithms
- Solve the Big Data Analytics problems using various technologies and tools

List of Lab Tasks

Task 1: Configure apache hadoop environment: To understand distributed and parallel computing environment.

Task 2: Hadoop file management: Adding files and directories , Retrieving files , Deleting files.

Task 3: Word Count application: Map-Reduce program to understand Map-Reduce Paradigm.

Task 4: Semi structured data: Mapreduce program that mines a given semi structured dataset (weather data.)

Task 5: Graph Mining: Use Map-Reduce to find the shortest path between two people in a

social graph.

Task 6:NO-SQL database –Apcache Hbase: To set Hbase shell environment and to create tables, insert rows, display contents etc.

Task 7:Pig Latin scripts : To sort, group, join for a given dataset.

Task 8:Database manipulation using Hive: To create, alter, drop databases and views.

Task 9:Functions and indexes in Hive.

Task 10:Use Hive to Drop Functions and indexes.

III Year I I Semester		L	T	P	C
		0	3	1.5	
CLOUD COMPUTING AB					

- 1.Install Virtualbox/VMware Workstation with different flavours of linux or windows OS on top of windows7 or 8.
2. Install a C compiler in the virtual machine created using virtual box and execute Simple Programs.
3. Install Google App Engine. Create hello world app and other simple web applications using python/java.
4. Use GAE launcher to launch the web applications.
5. Simulate a cloud scenario using CloudSim and run a scheduling algorithm that is not present in CloudSim.
6. Find a procedure to transfer the files from one virtual machine to another virtual machine.

7. Find a procedure to launch virtual machine using trystack (Online Openstack Demo Version)
8. Install Hadoop single node cluster and run simple applications like wordcount.

III Year II Semester	L	T	P	C
	3	0	0	3
CRYPTOGRAPHY NETWORK SECURITY LAB				

Course Objectives:

- To learn basic understanding of cryptography, how it has evolved, and some key encryption techniques used today.
- To understand and implement encryption and decryption using Ceaser Cipher, Substitution Cipher, Hill Cipher.

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

- Apply the knowledge of symmetric cryptography to implement encryption and decryption using Ceaser Cipher, Substitution Cipher, Hill Cipher
- Demonstrate the different algorithms like DES, BlowFish, and Rijndael, encrypt the text "Hello world" using Blowfish Algorithm.
- Analyze and implement public key algorithms like RSA, Diffie-Hellman Key Exchange mechanism, the message digest of a text using the SHA-1 algorithm

List of Experiments:

1. Write a C program that contains a string (char pointer) with a value 'Hello World'. The

- program
should XOR each character in this string with 0 and displays the result.
2. Write a C program that contains a string (char pointer) with a value 'Hello World'. The program should AND or and XOR each character in this string with 127 and display the result
 3. Write a Java program to perform encryption and decryption using the following algorithms:
 - a) Ceaser Cipher
 - b) Substitution Cipher
 - c) Hill Cipher
 4. Write a Java program to implement the DES algorithm logic
 5. Write a C/JAVA program to implement the BlowFish algorithm logic
 6. Write a C/JAVA program to implement the Rijndael algorithm logic.
 7. Using Java Cryptography, encrypt the text "Hello world" using BlowFish. Create your own key using Java key tool.
 8. Write a Java program to implement RSA Algorithm
 9. Implement the Diffie-Hellman Key Exchange mechanism using HTML and JavaScript. Consider the end user as one of the parties (Alice) and the JavaScript application as other party (bob).
 10. Calculate the message digest of a text using the SHA-1 algorithm in JAVA.

III B.Tech II Sem	L	T	P	C
	0	0	4	2
MEANSTACKTECHNOLOGIES-MODULEI-HTML5,JAVASCRIPT,NODE.JS EXPRESS.JSANDTYPESCRIPT (Skill Oriented Course - IV)				

Course Outcomes:

At the end of the Course, Student will be able to:

- Develop professional web pages of an application using HTML elements like lists, navigations, tables, various form elements, embedded media which includes images, audio, video and CSS Styles.
- Utilize JavaScript for developing interactive HTML web pages and validate form data.
- Build a basic web server using Node.js and also working with Node Package Manager (NPM).
- Build a web server using Express.js
- Make use of Typescript to optimize JavaScript code by using the concept of strict type checking.

List of Exercises

1.a	CourseName:HTML5-The Language
	ModuleName:Case-insensitivity,Platform-independency,DOCTYPEDeclaration, TypesofElements,HTMLElements-Attributes,MetadataElement

	IncludetheMetadataelementinHomepage.htmlforprovidingdescriptionas "IEKart'sisanonlineshoppingwebsitethatsellsgoodsinretail.Thiscompanydeals withvariouscategorieslikeElectronics,Clothing,Accessoriesetc. https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_28320667711144660000_shared?collectionId=lex_17739732834840810000_shared&collectionType=Course
1.b	CourseName: HTML5-The Language
	ModuleName: SectioningElements
	EnhancetheHomepage.htmlofIEKart'sShoppingApplicationbyaddingappropriate sectioningelements. https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_6372291347110857000_shared?collectionId=lex_17739732834840810000_s_hared&collectionType=Course
1.c	CourseName: HTML5-The Language
	ModuleName: ParagraphElement,DivisionandSpanElements,ListElement
	Makeuseofappropriategroupingelementssuchaslistitemsto "AboutUs"pageof IEKart'sShoppingApplication https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_32785192040894940000_shared?collectionId=lex_17739732834840810000_shared&collectionType=Course
1.d	CourseName: HTML5-The Language
	ModuleName: LinkElement
	Link "Login","SignUp"and"Trackorder"to"Login.html","SignUp.html"and"Track.html"p agerespectively.BookmarkeachcategorytoitsdetailsofIEKart's Shoppingapplication. https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_15515105953273338000_shared?collectionId=lex_17739732834840810000_shared&collectionType=Course

1.e	CourseName: HTML5-The Language
	ModuleName: Character Entities
	Addthe©symbolintheHomepagefooterofIEKart'sShopping application. https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_547667376938096260_shared?collectionId=lex_17739732834840810000_shared&collectionType=Course
1.f	CourseName: HTML5-The Language
	ModuleName: HTML5GlobalAttributes
	Addtheglobalattributessuchascontenteditable,spellcheck,idetc.toenhancethe SignupPagefunctionalityofIEKart'sShoppingapplication. https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_28723566050321920000_shared?collectionId=lex_17739732834840810000_shared&collectionType=Course
2.a	CourseName: HTML5-The Language
	ModuleName: CreatingTableElements,TableElements:Colspan/Rowspan Attributes, border, cellspacing, cellpadding attributes
	EnhancethedetailspageofIEKart'sShoppingapplicationbyaddingatableelement todisplaytheavailablemobile/anyinventories. https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_auth_013168035284033536113_shared?collectionId=lex_17739732834840810000_shared&collectionType=Course
2.b	CourseName: HTML5-The Language
	ModuleName: CreatingFormElements,ColorandDatePickers,SelectandDatalist Elements

	Using the form elements create Sign up page for IEKart's Shopping application. https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_13975270903118459000_shared?collectionId=lex_17739732834840810000_shared&collectionType=Course
2.c	CourseName: HTML5-The Language
	ModuleName: InputElements-Attributes
	Enhance Sign up page functionality of IEKart's Shopping application by adding attributes to input elements.
	https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_14048414537062347000_shared?collectionId=lex_17739732834840810000_shared&collectionType=Course
2.d	CourseName: HTML5-The Language
	ModuleName: Media, IFrame
	Add media content in a frame using audio, video, iframe elements to the Homepage of IEKart's Shopping application.
	https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_30738402225794945000_shared?collectionId=lex_17739732834840810000_shared&collectionType=Course
3.a	CourseName: Javascript
	ModuleName: Type of Identifiers
	Write a JavaScript program to find the area of a circle using radius (var and let-reassign and observe the difference with var and let) and PI (const)
	https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_auth_013053264414818304732_shared?collectionId=lex_181096983663

	32810000_shared&collectionType=Course
3.b	CourseName: Javascript
	ModuleName: Primitive and Non Primitive Data Types
	Write JavaScript code to display the movie details such as movie name, starring, language, and ratings. Initialize the variables with values of appropriate types. Use template literals wherever necessary.
	https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_21528322245232402000_shared?collectionId=lex_18109698366332810000_shared&collectionType=Course
3.c	CourseName: Javascript
	ModuleName: Operators and Types of Operators
	Write JavaScript code to book movie tickets online and calculate the total price, considering the number of tickets and price per ticket as Rs. 150. Also, apply a festive season discount of 10% and calculate the discounted amount.
	https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_13808338384481720000_shared?collectionId=lex_18109698366332810000_shared&collectionType=Course
3.d	CourseName: Javascript
	ModuleName: Types of Statements, Non-Conditional Statements, Types of Conditional Statements, if Statements, switch Statements
	Write JavaScript code to book movie tickets online and calculate the total price based on the 3 conditions: (a) If seat to be booked are not more than 2, the cost per ticket remains Rs. 150. (b) If seats are 6 or more, booking is not allowed. (c) If se
	https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_16257498471333610000_shared?collectionId=lex_18109698366332810000_shared&collectionType=Course
3.e	CourseName: Javascript

	ModuleName: TypesofLoops
	WriteaJavaScriptcodetobookmovieticketsonlineandcalculatethetotalprice basedonthe3conditions:(a)Ifseatstobebookedarenotmorethan2,thecostper ticketremains Rs.150.(b) Ifseatsare6ormore,bookingisnotallowed.(c)If
	https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_6238536888292970000_shared?collectionId=lex_1810969836633281000_0_shared&collectionType=Course
4.a	CourseName: Javascript
	ModuleName: TypesofFunctions,DeclaringandInvokingFunction,Arrow Function,FunctionParameters,NestedFunction,Built-inFunctions,VariableScope inFunctions
	WriteaJavaScriptcodetobookmovieticketsonlineandcalculatethetotalprice based on the 3 conditions: (a) If seats to be booked are not more than 2, the costperticket remains Rs. 150.(b) Ifseats are 6 or more, booking is not allowed.(c) If
	https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_15455199570613326000_shared?collectionId=lex_1810969836633281000_00_shared&collectionType=Course
4.b	CourseName: Javascript
	ModuleName: WorkingWithClasses,CreatingandInheritingClasses
	CreateanEmployeeclassextending fromabaseclassPerson.Hints:(i) Createaclass Personwithnameandageasattributes.(ii)Addaconstructortoinitializethevalues

	(iii)CreateaclassEmployeeextendingPersonwithadditionalattributesrole
	https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_auth_012599811117760512458_shared?collectionId=lex_18109698366332810000_shared&collectionType=Course
4.c	CourseName: Javascript
	ModuleName: In-builtEventsandHandlers
	WriteaJavaScriptcodetobookmovieticketsonlineandcalculatethetotalprice based on the 3 conditions: (a) If seats to be booked are not more than 2, the costperticket remains Rs. 150. (b) Ifseats are 6or more, booking is not allowed. (c) Ifse
	https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_4192188372573027000_shared?collectionId=lex_1810969836633281000_0_shared&collectionType=Course
4.d	CourseName: Javascript
	ModuleName: WorkingwithObjects,TypesofObjects,CreatingObjects, CombiningandcloningObjectsusingSpreadoperator,DestructuringObjects, BrowserObjectModel,DocumentObject Model
	Ifauserclicksonthegivenlink,theyshouldseeanemptycone,adifferentheading, andadifferentmessageandadifferentbackgroundcolor.If userclicksagain,they should see a re-filled cone, a different heading, a different message, and a diffe
	https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_13197025862804100000_shared?collectionId=lex_18109698366332810000_shared&collectionType=Course
5.a	CourseName: Javascript
	ModuleName: CreatingArrays,DestructuringArrays,AccessingArrays,Array Methods
	Createanarrayofobjectshavingmoviedetails.Theobjectshouldincludethemovie name, starring, language, and ratings. Render the details of movies on the page using the array.
	https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_auth_013053270191734784711_shared?collectionId=lex_18109698366332810000_shared&collectionType=Course
5.b	CourseName: Javascript

	<p>ModuleName: Introduction to Asynchronous Programming, Callbacks, Promises, Async and Await, Executing Network Requests using Fetch API</p> <p>Simulate a periodic stock price change and display on the console. Hints: (i) Create a method which returns a random number - use Math.random, floor and other methods to return a rounded value. (ii) Invoke the method for every three seconds and stop when</p> <p>https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_auth_012599811633905664460_shared?collectionId=lex_18109698366332810000_shared&collectionType=Course</p>
5.c	<p>CourseName: Javascript</p> <p>ModuleName: Creating Modules, Consuming Modules</p> <p>Validate the user by creating a login module. Hints: (i) Create a file login.js with a User class. (ii) Create a validate method with username and password as arguments. (iii) If the username and password are equal it will return "Login Successful" else</p> <p>https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_auth_013052857053585408667_shared?collectionId=lex_181096983663</p>

	<p>32810000_shared&collectionType=Course</p>
6.a	<p>CourseName: Node.js</p> <p>ModuleName: How to use Node.js</p> <p>Verify how to execute different functions successfully in the Node.js platform.</p> <p>https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_19002830632103186000_shared?collectionId=lex_32407835671946760000_shared&collectionType=Course</p>
6.b	<p>CourseName: Node.js</p> <p>ModuleName: Create a web server in Node.js</p> <p>Write a program to show the workflow of JavaScript code executable by creating a web server in Node.js.</p> <p>https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_28177338996267815000_shared?collectionId=lex_32407835671946760000_shared&collectionType=Course</p>
6.c	<p>CourseName: Node.js</p> <p>ModuleName: Modular programming in Node.js</p> <p>Write a Node.js module to show the workflow of modularization of Node application.</p> <p>https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_28865394191004004000_shared?collectionId=lex_32407835671946760000_shared&collectionType=Course</p>
6.d	<p>CourseName: Node.js</p> <p>ModuleName: Restarting Node Application</p> <p>Write a program to show the workflow of restarting a Node application.</p> <p>https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_9174073856000159000_shared?collectionId=lex_32407835671946760000_shared&collectionType=Course</p>
6.e	<p>CourseName: Node.js</p> <p>ModuleName: File Operations</p> <p>Create a text file src.txt and add the following data to it. Mongo, Express, Angular, Node.</p> <p>https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_33376440180246100000_shared?collectionId=lex_32407835671946760000_shared&collectionType=Course</p>
7.a	<p>CourseName: Express.js</p>

	ModuleName: Definingaroute,HandlingRoutes,RouteParameters,Query Parameters
	ImplementroutingfortheAdventureTrailsapplicationbyembeddingthenecessary codeintheroutes/route.jsfile.
	https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_29394215542149950000_shared?collectionId=lex_324078356719467600_00_shared&collectionType=Course
7.b	CourseName: Express.js
	ModuleName: HowMiddlewareworks,ChainingofMiddlewares,Typesof Middlewares
	InmyNotesapplication:(i)wewanttohandlePOSTsubmissions.(ii)display customizederrormessages.(iii)performlogging.
	https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_13930661312009580000_shared?collectionId=lex_324078356719467600

	00_shared&collectionType=Course
7.c	CourseName: Express.js
	ModuleName: ConnectingtoMongoDBwithMongoose,ValidationTypesand Defaults
	WriteaMongoosesthematoconnectwithMongoDB.
	https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_auth_013035588775485440691_shared?collectionId=lex_32407835671946760000_shared&collectionType=Course
7.d	CourseName: Express.js
	ModuleName: Models
	Writeaprogramto wraptheSchemaintoamodelobject.
	https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_auth_013035593896869888662_shared?collectionId=lex_32407835671946760000_shared&collectionType=Course
8.a	CourseName: Express.js
	ModuleName: CRUD Operations
	Writeaprogramto performvariousCRUD(Create-Read-Update-Delete)operations usingMongooselibraryfunctions.
	https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_auth_013035684270129152696_shared?collectionId=lex_32407835671946760000_shared&collectionType=Course
8.b	CourseName: Express.js
	ModuleName: APIDevelopment
	In the myNotes application, include APIs based on the requirements provided. (i) API should fetchthe details ofthe notes based ona notesID which is provided inthe URL. Test URL - http://localhost:3000/notes/7555(ii) API should update the details bas
	https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_auth_013035745250975744755_shared?collectionId=lex_32407835671946760000_shared&collectionType=Course
8.c	CourseName: Express.js
	ModuleName: WhySessionmanagement,Cookies
	Writeaprogramto explainsessionmanagementusing cookies.
	https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_24299316914857090000_shared?collectionId=lex_32407835671946760000_shared&collectionType=Course
8.d	CourseName: Express.js

	ModuleName: Sessions
	Writeaprogramtoexplainsessionmanagementusing sessions.
	https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_905413034723449100_shared?collectionId=lex_32407835671946760000_shared&collectionType=Course
8.e	CourseName: Express.js
	ModuleName: Whyand WhatSecurity,HelmetMiddleware
	ImplementsecurityfeaturesinmyNotesapplication
	https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_31677453061177940000_shared?collectionId=lex_32407835671946760000_shared&collectionType=Course

9.a	CourseName: Typescript
	ModuleName: BasicsofTypeScript
	Onthepage,display thepriceofthemobile-basedinthreedifferentcolors.Insteadof usingthenumberinourcode,representthembystringvalueslikeGoldPlatinum, PinkGold, SilverTitanium.
	https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_28910354929502245000_shared?collectionId=lex_943623311651267800_0_shared&collectionType=Course
9.b	CourseName: Typescript
	ModuleName: Function
	Defineanarrowfunctioninsidetheeventhandlertofiltertheproductarraywiththe selected product object using the productId received bythe function. Pass the selected product object to the next screen.
	https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_10783156469383723000_shared?collectionId=lex_943623311651267800_0_shared&collectionType=Course
9.c	CourseName: Typescript
	ModuleName: ParameterTypesandReturnTypes
	Considerthatdeveloperneedstodeclareafunction-getMobileByVendorwhich acceptsstringas inputparameterandreturnsthelistofmobiles.
	https://infyspringboard.onwingspan.com/web/en/viewer/hands-on/lex_auth_012712912427057152901_shared?collectionId=lex_9436233116512678000_shared&collectionType=Course
9.d	CourseName: Typescript
	ModuleName: ArrowFunction
	Considerthatdeveloperneeds todeclareamanufacturer's arrayholding4objects with id and price as a parameter and needs to implement anarrow function - myfunctionto populate the id parameter of manufacturers array whose price is greater than or equ
	https://infyspringboard.onwingspan.com/web/en/viewer/hands-on/lex_auth_012712910875500544904_shared?collectionId=lex_9436233116512678000_shared&collectionType=Course
9.e	CourseName: Typescript
	ModuleName: OptionalandDefaultParameters
	Declareafunction-getMobileByManufacturerwithtwoparametersnamely manufacturerandid,wheremanufacturervalueshouldpassedasSamsungandid parameter should be optional while invoking the function, if id is passed as 101 thenthis function shoul
	https://infyspringboard.onwingspan.com/web/en/viewer/hands-on/lex_auth_012712914940641280906_shared?collectionId=lex_9436233116512678000_shared&collectionType=Course

10.a	CourseName: Typescript
	ModuleName: RestParameter
	Implementbusiness logicforaddingmultipleProductvaluesintoacartvariablewhich istypeofstring array.
	https://infyspringboard.onwingspan.com/web/en/viewer/hands-on/lex_auth_012712921860915200909_shared?collectionId=lex_9436233116512678000_shared&collectionType=Course

10.b	CourseName: Typescript
	ModuleName: CreatinganInterface
	Declareaninterfacenamed-ProductwithtwopropertieslikeproductIdand productNameewithanumberandstringdatatypeandneedtoimplementlogicto populate the Product details.
	https://infyspringboard.onwingspan.com/web/en/viewer/hands-on/lex_auth_012712925244276736910_shared?collectionId=lex_9436233116512678000_shared&collectionType=Course

10.c	CourseName: Typescript
	ModuleName: Duck Typing
	Declareaninterfacenamed-ProductwithtwopropertieslikeproductIdand productNameewiththenumberandstringdatatypeandneedtoimplementlogicto populate the Product details.
	https://infyspringboard.onwingspan.com/web/en/viewer/hands-on/lex_auth_012712925995458560912_shared?collectionId=lex_9436233116512678000_shared&collectionType=Course

10.d	CourseName: Typescript
	ModuleName: FunctionTypes
	Declareaninterface withfunctiontypeandaccess itsvalue.
	https://infyspringboard.onwingspan.com/web/en/viewer/hands-on/lex_auth_012712948945346560918_shared?collectionId=lex_9436233116512678000_shared&collectionType=Course

11.a	CourseName: Typescript
	ModuleName: ExtendingInterfaces
	DeclareaproductListinterfacewhichextendspropertiesfromtwootherdeclared interfaceslikeCategory,Productaswellasimplementationtocreateavariablenameofthis interfacetype.
	https://infyspringboard.onwingspan.com/web/en/viewer/hands-on/lex_auth_012712951652139008920_shared?collectionId=lex_9436233116512678000_shared&collectionType=Course

11.b	CourseName: Typescript
	ModuleName: Classes
	ConsidertheMobileCartapplication,CreateobjectsoftheProductclassandplace themintotheproductlistarray.
	https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_3705824317381604400_shared?collectionId=lex_9436233116512678000_shared&collectionType=Course

11.c	CourseName: Typescript
	ModuleName: Constructor
	Declareaclassnamed-Productwiththebelow-mentioneddeclarations:(i)productId as number property(ii) Constructorto initialize this value (iii) getProductIdmethodto return the message "Product id is <<id value>>".

	https://infyspringboard.onwingspan.com/web/en/viewer/hands-on/lex_auth_012712954616782848927_shared?collectionId=lex_9436233116512678000_shared&collectionType=Course
11.d	CourseName: TypeScript
	ModuleName: AccessModifiers
	Create a Product class with 4 properties namely productId, productName,

	productPrice,productCategorywithprivate,public,static,andprotectedaccess modifiersandaccessingthemthroughGadgetclassanditsmethods.
	https://infyspringboard.onwingspan.com/web/en/viewer/hands-on/lex_auth_012712953517170688931_shared?collectionId=lex_9436233116512678000_shared&collectionType=Course
12.a	CourseName: TypeScript
	ModuleName: Propertiesand Methods
	CreateaProductclasswith4propertiesnamelyproductIdandmethodsto setProductId()andgetProductId().
	https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_9356738095572543000_shared?collectionId=lex_9436233116512678000_shared&collectionType=Course
12.b	CourseName: TypeScript
	ModuleName: Creatingandusing Namespaces
	CreateanamespacecalledProductUtilityandplacetheProductclassdefinitioninit. ImporttheProductclassinsideproductlistfileanduse it.
	https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_20787271128051925000_shared?collectionId=lex_9436233116512678000_shared&collectionType=Course
12.c	CourseName: TypeScript
	ModuleName: Creatingandusing Modules
	ConsidertheMobileCartapplicationwhichisdesignedaspartofthefunctionsina module to calculate the total price of the product using the quantity and price valuesand assign it to a totalPrice variable.
	https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_24788158187785620000_shared?collectionId=lex_9436233116512678000_shared&collectionType=Course
12.d	CourseName: TypeScript
	ModuleName: WhatisGenerics,WhatareTypeParameters,GenericFunctions, GenericConstraints
	Createagenericarrayandfunctiontosortnumbersaswellasstring values.
	https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_446287045482942800_shared?collectionId=lex_9436233116512678000_shared&collectionType=Course

Softwareconfiguration andinstallation

1. HTML&JavaScript

- SimpleeditorssuchasNotepadorgoforIDESlikeVisualStudioCode(recommended),Eclipse etc. which makes coding easier.
- And,to executeapplication, youcanuseanycommonlyused browsersuchasGoogle Chrome(recommended), Mozilla Firefox etc
- Setup details: [EnvironmentalSetup for HTML5 - Viewer Page | Infosys Springboard\(onwingspan.com\)](#)
- [Environment Setup:Internal- ViewerPage|InfosysSpringboard\(onwingspan.com\)](#)

2. NodeJS

Download **Node.js** from the official site

[Setup details: How to use Node.js - Viewer Page | Infosys Springboard \(onwingspan.com\)](#)

3. Typescript

[Installing TypeScript - Internal - Viewer Page | Infosys Springboard \(onwingspan.com\)](#)

Text Books:

1. Programming the World Wide Web, 7th Edition, Robert W. Sebesta, Pearson.
2. Pro Mean Stack Development, 1st Edition, Elad Elrom, Apress O'Reilly.
3. Full Stack JavaScript Development with MEAN, Colin J. Hrig, Adam Bretz, 1st Edition, SitePoint, SitePoint Pty. Ltd., O'Reilly Media.

Reference Books:

1. Web Technologies, HTML, JavaScript, PHP, Java, JSP, XML and AJAX, Blackbook, 1st Edition, Dream Tech.
2. An Introduction to Web Design, Programming, 1st Edition, Paul S. Wang, Sanda S. Katila, Cengage Learning.

Web Links:

1. https://infyspringboard.onwingspan.com/en/app/toc/lex_17739732834840810000_shared/overview (HTML5)
2. https://infyspringboard.onwingspan.com/en/app/toc/lex_18109698366332810000_shared/overview (JavaScript)
3. https://infyspringboard.onwingspan.com/en/app/toc/lex_32407835671946760000_shared/overview (Node.js & Express.js)
4. https://infyspringboard.onwingspan.com/en/app/toc/lex_9436233116512678000_shared/overview (Typescript)

ES – II

IV Year – I Semester		L	T	P	C
		3	0	0	3
NEURAL NETWORKS AND SOFT COMPUTING (Professional Elective-III)					

Course Objectives: The main objective of the course is

- To introduce the foundations of Artificial Neural Networks.
- To acquire the knowledge on Soft Computing Concepts.
- To learn various types of Genetic algorithms and its applications.
- To gain knowledge to apply optimization strategies.

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

- Understand the concepts of Artificial intelligence and soft computing techniques
- Analyze the concepts of Neural Networks and select the Learning Networks in modeling real world systems.
- Implement the concepts of Fuzzy reasoning and concepts of Genetic algorithm and its applications to soft computing.
- Classify Biologically inspired algorithm such as neural networks, genetic algorithms, ant colony optimization, and bee colony optimization.
- Design hybrid system incorporating neural network, genetic algorithms, fuzzy systems.

UNIT I: Soft Computing and Artificial Intelligence: Introduction of Soft Computing, Soft Computing vs. Hard Computing, Various Types of Soft Computing Techniques, Applications of Soft Computing, AI Search Algorithm, Predicate Calculus, Rules of Inference, Semantic Networks, Frames, Objects, Hybrid Models.

UNIT II: Artificial Neural Networks and Paradigms: Introduction to Neuron Model, Neural Network Architecture, Learning Rules, Perceptrons, Single Layer Perceptrons, Multilayer Perceptrons, Back propagation Networks, Kohonen's self organizing networks, Hopfield network, Applications of NN.

UNIT III: Fuzzy Logic: Introduction, Fuzzy sets and Fuzzy reasoning, Basic functions on fuzzy sets, relations, rule based models and linguistic variables, fuzzy controls, Fuzzy decision making, applications of fuzzy logic.

UNIT IV: Genetic Algorithms and Swarm Optimizations: Introduction, Genetic Algorithm, Fitness Computations, Cross Over, Mutation, Evolutionary Programming, Classifier Systems, Genetic Programming Parse Trees, Variants of GA, Applications, Ant Colony Optimization, Particle Swarm Optimization, Artificial Bee Colony Optimization.

UNIT V: Hybrid Systems: Neuro fuzzy hybrid systems, Adaptive neuro fuzzy inference systems, Fuzzy backpropagation network, Genetic neuro hybrid system, Genetic algorithm based backpropagation network, Genetic-fuzzy hybrid systems.

Text Books:

1. Simon S. Haykin, Neural Networks, Prentice Hall, 2nd edition.
2. S. Rajasekaran & G. A. Vijayalakshmi Pai "Neural Networks, Fuzzy Logic and Genetic Algorithms: Synthesis & Applications", PHI, 2003.

Reference Books:

1. S. N. Sivanandam & S. N. Deepa "Principles of Soft Computing" Wiley – India, 2nd Edition, 2007.
2. Jang J.S.R., Sun C.T. and Mizutani E, "Neuro-Fuzzy and Soft computing", Prentice Hall, 1998.
3. Jacek M. Zurada, Introduction to Artificial Neural Systems, Jaico Publishing House, 1994
4. Zimmermann, "Fuzzy Set Theory and its Application", 3rd Edition.
5. D.E. Goldberg, "Genetic Algorithms: Search, Optimization and Machine Learning", AddisonWesley, N.Y, 1989.
6. Timothy J. Ross, "Fuzzy Logic with Engineering Applications", McGraw Hill, 3rd edition 2009.

IV Year – I Semester	L	T	P	C
	3	0	0	3
AD-HOC AND SENSOR NETWORKS (Professional Elective-III)				

Course Objectives:

From the course the student will learn

- Architect sensor networks for various application setups.
- Devise appropriate data dissemination protocols and model links cost.
- Understanding of the fundamental concepts of wireless sensor networks and has a basic knowledge of the various protocols at various layers.
- Evaluate the performance of sensor networks and identify bottlenecks.

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

- Evaluate the principles and characteristics of mobile ad hoc networks (MANETs) and what distinguishes them from infrastructure-based networks.
- Determine the principles and characteristics of wireless sensor networks.
- Discuss the challenges in designing MAC, routing and transport protocols for wireless ad-hoc sensor networks.
- Illustrate the various sensor network Platforms, tools and applications.
- Demonstrate the issues and challenges in security provisioning and also familiar with the mechanisms for implementing security and trust mechanisms in MANETs and WSNs.

UNIT I: Introduction to Ad Hoc Wireless Networks- Cellular and Ad Hoc Wireless Networks, Characteristics of MANETs, Applications of MANETs, Issues and Challenges of MANETs, Ad Hoc Wireless Internet, MAC protocols for Ad hoc Wireless Networks-Issues, Design Goals and Classification of the MAC Protocols.

UNIT II: Routing Protocols for Ad Hoc Wireless Networks- Issues in Designing a Routing Protocol, Classifications of Routing Protocols, Topology-based versus Position-based Approaches, Issues and design goals of a Transport layer protocol, Classification of Transport layer solutions, TCP over Ad hoc Wireless Networks, Solutions for TCP over Ad Hoc Wireless Networks, Other Transport layer protocols.

UNIT III: Security protocols for Ad hoc Wireless Networks- Security in Ad hoc Wireless Networks, Network Security Requirements, Issues and Challenges in Security Provisioning, Network Security Attacks, Key Management, Secure Routing in Ad hoc Wireless Networks, Cooperation in MANETs, Intrusion Detection Systems.

UNIT IV: Basics of Wireless Sensors and Applications- The Mica Mote, Sensing and Communication Range, Design Issues, Energy Consumption, Clustering of Sensors, Applications, Data Retrieval in Sensor Networks- Classification of WSNs, MAC layer, Routing layer, Transport layer, High-level application layer support, Adapting to the inherent dynamic nature of WSNs.

UNIT V: Security in WSNs- Security in WSNs, Key Management in WSNs, Secure Data Aggregation in WSNs, Sensor Network Hardware-Components of Sensor Mote, Sensor Network Operating Systems– TinyOS, LA-TinyOS, SOS, RETOS, Imperative Language-nesC, **Dataflow Style Language**-TinyGALS, Node-Level Simulators, NS-2 and its sensor network extension, TOSSIM.

Text Books:

1. Ad Hoc Wireless Networks – Architectures and Protocols, 1st edition, C. Siva Ram Murthy, B. S. Murthy, Pearson Education, 2004
2. Ad Hoc and Sensor Networks – Theory and Applications, 2nd edition *Carlos Corderio Dharma P. Aggarwal*, World Scientific Publications / Cambridge University Press, March 2006

Reference Books:

1. Wireless Sensor Networks: An Information Processing Approach, 1st edition, *Feng Zhao, Leonidas Guibas*, Elsevier Science imprint, Morgan Kaufman Publishers, 2005, rp2009
2. Wireless Ad hoc Mobile Wireless Networks – Principles, Protocols and Applications, 1st edition, Subir Kumar Sarkar, et al., Auerbach Publications, Taylor & Francis Group, 2008
3. Ad hoc Networking, 1st edition, *Charles E. Perkins*, Pearson Education, 2001
4. Wireless Ad hoc Networking, 1st edition, *Shih-Lin Wu, Yu-Chee Tseng*, Auerbach Publications, Taylor & Francis Group, 2007
5. Wireless Sensor Networks – Principles and Practice, 1st edition, Fei Hu, Xiaojun Cao, An Auerbach book, CRC Press, Taylor & Francis Group, 2010

IV Year – I Semester		L	T	P	C
		3	0	0	3
CYBER SECURITY & FORENSICS (Professional Elective-III)					

Course Objectives:

The aim of the course is to

- identify security risks and take preventive steps
- understand the forensics fundamentals
- understand the evidence capturing process
- understand the preservation of digital evidence

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

- Explain the Cybercrime Fundamentals
- Describe the types of attacks on networks
- Analyze various tools available for Cybercrime Investigation
- Explain the Computer Forensics and Investigation Fundamentals and tools
- Analyze the legal perspectives of Cybercrime

UNIT I: Introduction to Cybercrime: Introduction, Cybercrime: Definition and Origins of the Word, Cybercrime and Information Security, Cybercriminals, Classifications of Cybercrime, Cyberstalking, Cybercafe and Cybercrimes, Botnets. Attack Vector, Proliferation of Mobile and Wireless Devices, Security Challenges Posed by Mobile Devices, Attacks on Mobile/Cell Phones, Network and Computer Attacks.

UNIT II: Tools and Methods : Proxy Servers and Anonymizers, Phishing, Password Cracking, Keyloggers and Spywares, Virus and Worms, Trojan Horses and Backdoors, Steganography, Sniffers, Spoofing, Session Hijacking Buffer over flow, DoS and DDoS Attacks, SQL Injection, Buffer Overflow, Attacks on Wireless Networks, Identity Theft (ID Theft), Foot Printing and Social Engineering, Port Scanning, Enumeration.

UNIT III: Cyber Crime Investigation: Introduction, Investigation Tools, eDiscovery, Digital Evidence Collection, Evidence Preservation, E-Mail Investigation, E-Mail Tracking, IP Tracking, E-Mail Recovery, Hands on Case Studies. Encryption and Decryption Methods, Search and Seizure of Computers, Recovering Deleted Evidences, Password Cracking.

UNIT IV: Computer Forensics and Investigations: Understanding Computer Forensics, Preparing for Computer Investigations. Current Computer Forensics Tools: Evaluating Computer Forensics Tools, Computer Forensics Software Tools, Computer Forensics Hardware Tools, Validating and Testing Forensics Software, Face, Iris and Fingerprint Recognition, Audio Video Analysis, Windows System Forensics, Linux System Forensics, Graphics and Network Forensics, E-mail Investigations, Cell Phone and Mobile Device Forensics.

UNIT V: Cyber Crime Legal Perspectives: Introduction, Cybercrime and the Legal Landscape around the World, The Indian IT Act, Challenges to Indian Law and Cybercrime Scenario in India, Consequences of Not Addressing the Weakness in Information Technology Act, Digital Signatures and the Indian IT Act, Amendments to the Indian IT Act, Cybercrime and Punishment, Cyberlaw, Technology and Students: Indian Scenario.

Text Books:

1. Sunit Belapure Nina Godbole "Cyber Security: Understanding Cyber Crimes, ComputerForensics and Legal Perspectives", WILEY, 2011.
2. Nelson Phillips and Enfinger Steuart, "Computer Forensics and Investigations", CengageLearning, New Delhi, 2009.

Reference Books:

1. Michael T. Simpson, Kent Backman and James E. Corley, "Hands on Ethical Hacking andNetwork Defence", Cengage, 2019.
2. Computer Forensics, Computer Crime Investigation by John R. Vacca, Firewall Media, NewDelhi.
3. Alfred Basta, Nadine Basta,Mary Brown and Ravinder Kumar "Cyber Security and CyberLaws" , Cengage,2018.

E-Resources:

1. CERT-In Guidelines- <http://www.cert-in.org.in/>
2. <https://www.coursera.org/learn/introduction-cybersecurity-cyber-attacks> [Online Course]
3. <https://computersecurity.stanford.edu/free-online-videos> [Free Online Videos]
4. Nickolai Zeldovich. 6.858 Computer Systems Security. Fall 2014. Massachusetts Institute of Technology: MIT OpenCourseWare, <https://ocw.mit.edu> License: Creative Commons BY-NC- SA.

IV Year – I Semester		L	T	P	C
		3	0	0	3
DEEP LEARNING TECHNIQUES (Professional Elective-IV)					

Course Objectives: The main objective of the course is to make students:

- Learn deep learning methods for working with sequential data,
- Learn deep recurrent and memory networks,
- Learn deep Turing machines,
- Apply such deep learning mechanisms to various learning problems.
- Know the open issues in deep learning, and have a grasp of the current research directions.

Course Outcomes: After the completion of the course, student will be able to

- Demonstrate the fundamental concepts learning techniques of Artificial Intelligence, Machine Learning and Deep Learning.
- Discuss the Neural Network training, various random models.
- Explain the Techniques of Keras, TensorFlow, Theano and CNTK
- Classify the Concepts of CNN and RNN
- Implement Interactive Applications of Deep Learning.

UNIT I:

Fundamentals of Deep Learning: Artificial Intelligence, History of Machine learning: Probabilistic Modeling, Early Neural Networks, Kernel Methods, Decision Trees, Random forests and Gradient Boosting Machines, **Fundamentals of Machine Learning:** Four Branches of Machine Learning, Evaluating Machine learning Models, Overfitting and Underfitting. **[Text Book 2]**

UNIT II: Introducing Deep Learning: Biological and Machine Vision, Human and Machine Language, Artificial Neural Networks, Training Deep Networks, Improving Deep Networks. **[Text Book3]**

UNIT III: Neural Networks: Anatomy of Neural Network, Introduction to Keras: Keras, TensorFlow, Theano and CNTK, Setting up Deep Learning Workstation, Classifying Movie Reviews: Binary Classification, Classifying newswires: Multiclass Classification. **[Text Book 2]**

UNIT IV:

Convolutional Neural Networks: Nerual Network and Representation Learning, Convolutional Layers, Multichannel Convolution Operation, **Recurrent Neural Networks:** Introduction to RNN, RNN Code, PyTorch Tensors: Deep Learning with PyTorch, CNN in PyTorch. **[Text Book 3]**

UNIT V:

Interactive Applications of Deep Learning: Machine Vision, Natural Language processing, Generative Adversarial Networks, Deep Reinforcement Learning. **[Text Book 1]**

Deep Learning Research: Autoencoders, Deep Generative Models: Boltzmann Machines Restricted Boltzmann Machines, Deep Belief Networks. **[Text Book 1]**

Text Books:

1. Deep Learning- Ian Goodfellow, Yoshua Bengio and Aaron Courville, MIT Press, 2016
2. Deep Learning with Python - Francois Chollet, Released December 2017, Publisher(s): Manning Publications, ISBN: 9781617294433
3. Deep Learning Illustrated: A Visual, Interactive Guide to Artificial Intelligence - Jon Krohn, Grant Beyleveld, Aglaé Bassens, Released September 2019, Publisher(s): Addison-Wesley Professional, ISBN: 9780135116821

4. Learning from Scratch - Seth Weidman, Released September 2019, Publisher(s): O'ReillyMedia, Inc., ISBN: 9781492041412

Reference Books:

1. Artificial Neural Networks, Yegnanarayana, B., PHI Learning Pvt. Ltd, 2009.
2. Matrix Computations, Golub, G.,H., and Van Loan,C.,F, JHU Press,2013.
3. Neural Networks: A Classroom Approach, Satish Kumar, Tata McGraw-Hill Education, 2004.

Web Link:

1. Swayam NPTEL: Deep Learning: https://onlinecourses.nptel.ac.in/noc22_cs22/preview

IV Year – I Semester	L	T	P	C
	3	0	0	3
SOCIAL NETWORKS & SEMANTIC WEB (Professional Elective-IV)				

Course Objectives:

- To learn Web Intelligence
- To learn Knowledge Representation for the Semantic Web
- To learn Ontology Engineering
- To learn Semantic Web Applications, Services and Technology
- To learn Social Network Analysis and semantic web

Course Outcomes: After the completion of the course, student will be able to

- Demonstrate social network analysis and measures.
- Analyze random graph models and navigate social networks data
- Apply the network topology and Visualization tools.
- Analyze the experiment with small world models and clustering models.
- Compare the application driven virtual communities from social network Structure.

UNIT I: Web Intelligence: Thinking and Intelligent Web Applications, The Information Age ,The World Wide Web, Limitations of Today's Web, The Next Generation Web, Machine Intelligence, Artificial Intelligence, Ontology, Inference engines, Software Agents, Berners-Lee www, Semantic Road Map, Logic on the semantic Web.

UNIT II: Knowledge Representation for the Semantic Web: Ontologies and their role in the semantic web, Ontologies Languages for the Semantic Web –Resource Description Framework(RDF) / RDF Schema, Ontology Web Language(OWL), UML, XML/XML Schema.

UNIT III: Ontology Engineering: Ontology Engineering, Constructing Ontology, Ontology Development Tools, Ontology Methods, Ontology Sharing and Merging, Ontology Libraries and Ontology Mapping, Logic, Rule and Inference Engines.

UNIT IV: Semantic Web Applications, Services and Technology: Semantic Web applications and services, Semantic Search, e-learning, Semantic Bioinformatics, Knowledge Base ,XML Based Web Services, Creating an OWL-S Ontology for Web Services, Semantic Search Technology, Web Search Agents and Semantic Methods,

UNIT V: Social Network Analysis and semantic web: What is social Networks analysis, development of the social networks analysis, Electronic Sources for Network Analysis – Electronic Discussion networks, Blogs and Online Communities, Web Based Networks, Building Semantic Web Applications with social network features.

Text Books:

1. Thinking on the Web – Berners Lee, Godel and Turing, Wiley inter science, 2008.
2. Social Networks and the Semantic Web, Peter Mika, Springer, 2007.

Reference Books:

1. Semantic Web Technologies, Trends and Research in Ontology Based Systems, J. Davies, R. Studer, P. Warren, John Wiley & Sons.
2. Semantic Web and Semantic Web Services -Liyang Lu Chapman and

Hall/CRCPublishers,(Taylor & Francis Group)

3. Information sharing on the semantic Web – Heiner Stuckenschmidt; Frank Van Harmelen, Springer Publications.
4. Programming the Semantic Web, T. Segaran, C. Evans, J. Taylor, O'Reilly, SPD.

IV Year – I Semester	L	T	P	C
	3	0	0	3
COMPUTER VISION (Professional Elective-IV)				

Course Objectives:

To introduce students the fundamentals of image formation; To introduce students the major ideas, methods, and techniques of computer vision and pattern recognition; To develop an appreciation for various issues in the design of computer vision and object recognition systems; and To provide the student with programming experience from implementing computer vision and object recognition applications.

Course Outcomes: After completing the course Student will be able to:

1. Identify basic concepts, terminology, theories, models and methods in the field of computer vision,
2. Describe known principles of feature detection and matching,
3. Describe basic methods of computer vision related to image stitching, photography like highdynamic range imaging and blur removal.
4. Suggest a design of a computer vision system for a 3D Reconstruction, Albedos, image basedrendering views and depths.

UNIT I: Introduction: Image Formation: Geometric Primitives and Transformation, Photometric Image Formation, Digital Camera, Image Processing: Point Operators, Linear Filtering, More Neighborhood Operators, Fourier Transforms, Pyramids and Wavelets, Geometric Transformations, Global Optimization.

UNIT II: Feature Detection and Matching: Points and Patches, Edges, Lines, Segmentation: Active Contours, Split and Merge, Mean Shift and Mode Finding, Normalized Cuts, Feature-Based Alignment: 2D and 3D Feature-based Alignment, Pose Estimation, Geometric Intrinsic Calibration.

UNIT III: Structure and Motion: Triangular, Two-frame Structure from Motion, Factorization, Bundle Adjustment, Constrained Structure and Motion, Dense Motion Estimation: Translation Alignment, Parametric Motion, Spline-based Motion, Optical Flow, Layered motion

UNIT IV: Image Stitching: Motion Models, Global Alignment, Composing, Computational Photography: Photometric Calibration, High Dynamic Range Imaging, Super-Resolution and Blur Removal, image Matting and Compositing, Texture Analysis and Synthesis.

UNIT V: 3D Reconstruction: Shape From X, Active Range Finding, Surface Representation, Point- based Representation, Volumetric Representation, Model-based Reconstruction, Recovering Texture Maps and Albedos, Image- based Rendering: View Interpolation, Layered Depth Images, Light Fields and Lumigraphs, Environment Mattes, Video-based Rendering.

Text Books:

1. Richard Szeliski, Computer Vision: Algorithms and Applications, Springer-Verlag London Limited,2011.
2. Simon J.D Prince, Computer Vision: Models, Learning and Inference, 1st Edition, 2012.

Reference Books:

1. Computer Vision - A modern approach, by D. Forsyth and J. Ponce, Prentice Hall Robot Vision, by B. K. P. Horn, McGraw-Hill.
2. Haralick& Shapiro, "Computer and Robot Vision", Vol II
3. G_erardMedioni and Sing Bing Kang "Emerging topics in computer vision"166

MOOCS-NPTL/ SWAYAM

IV Year – I Semester		L	T	P	C
		3	0	0	3
BLOCK-CHAIN TECHNOLOGIES (Professional Elective-V)					

Course Objectives:

To understand block chain technology and Cryptocurrency works

Course Outcomes:

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

- Demonstrate the block chain basics, Crypto currency
- To compare and contrast the use of different private vs. public block chain and use cases
- Design an innovative Bit coin Block chain and scripts, Block chain Science on varies coins
- Classify Permission Block chain and use cases – Hyper ledger, Corda
- Make Use of Block-chain in E-Governance, Land Registration, Medical Information Systems and others

UNIT I: Introduction: Introduction, basic ideas behind block chain, how it is changing the landscape of digitalization, introduction to cryptographic concepts required, Block chain or distributed trust, Currency, Cryptocurrency, How a Cryptocurrency works, Financial services, Bitcoin prediction markets.

UNIT II: Hashing, public key cryptosystems, private vs public block chain and use cases, Hash Puzzles, Extensibility of Block chain concepts, Digital Identity verification, Block chain Neutrality, Digital art, Block chain Environment

UNIT III: Introduction to Bitcoin : Bitcoin Block chain and scripts, Use cases of Bitcoin Blockchain scripting language in micropayment, escrow etc Downside of Bit coin mining, Block chain Science: Grid coin, Folding coin, Block chain Genomics, Bit coin MOOCs.

UNIT IV: Ethereum continued, IOTA, The real need for mining, consensus, Byzantine Generals Problem, and Consensus as a distributed coordination problem, Coming to private or permissioned block chains, Introduction to Hyper ledger, Currency, Token, Campus coin, Coin drop as a strategy for Public adoption, Currency Multiplicity, Demurrage currency

UNIT V: Technical challenges, Business model challenges, Scandals and Public perception, Government Regulations, Uses of Block chain in E-Governance, Land Registration, Medical Information Systems.

Text Books:

1. Blockchain Blue print for Economy by Melanie Swan

Reference Books:

1. Blockchain Basics: A Non-Technical Introduction in 25 Steps 1st Edition, by Daniel Drescher

IV Year – I Semester	L	T	P	C
	3	0	0	3
WIRELESS NETWORK SECURITY (Professional Elective-V)				

Course Objectives:

The objective of this course is to understand the importance of Wireless networks security and its application

Course Outcomes: After the completion of the course, student will be able to

- Explain the Threats in networks and provide Authentication to real time problems.
- Identify and investigate in-depth both early and contemporary threats to wireless networks security
- Ability to analyze and determine for any organization the database security requirements and appropriate solutions
- Determined IP Security Issues and solve real time problems.
- Build wireless Development Strategies in real time issues

UNIT I: Introduction to Wireless: History of Wireless Technologies, History of Wireless Security, State of the Wireless Security Industry, 2001 **Wireless Threats:** Uncontrolled Terrain, Communications Jamming, DoS Jamming, Injections and Modifications of Data, Man-in-the-Middle (MITM) Attack, Rogue Client, Rogue Network Access Points, Attacker Equipment,

UNIT II: Introduction to Wireless Security Protocols and Cryptography: Recovery the FUD, OSI Model, OSI Simplified, Internet Model, Wireless LAN Security Protocols, Cryptography, SSL/TLS, Secure Shell Protocols, Terminal Access and File Transfer, Port Forwarding a Word of Caution, Man-in-the-Middle of SSL/TLS and SSH, WTLS, WEP, 802.1x, IP Security

UNIT III:

Security Considerations to Wireless Devices: Wireless Device Security Issues, Physical Security, Information Leakage, Device Security Features, Application Security, Detailed Device Analysis, Laptops, Personal Digital Assistants (PDAS), Wireless Infrastructure **Wireless Technologies and Applications:** Introduction to Cellular Networks- FDMA, TDMA, CDMA, Spread Spectrum Primer, Analogy, TDMA Vs CDMA, PDC, Security Threats

UNIT IV:

Introduction to Wireless Data Networks: Cellular Digital Packet Data (CDPD), CDPD Architecture, CDPD Security, Mobitex- Mobitex Architecture, Mobitex Security Architecture, Security Issues, Gateway, Security Model **Wireless Standards and Technologies:** Current and Future Technologies- Infrared, Radio, Spread Spectrum, OFDM, Current and Future Standards- IEEE 802 Standards, ETSI, Home RF, Ultra-wide band Radio (UWB)

UNIT V: Wireless Deployment Strategies: Implementing Wireless LAN's- Security Considerations Common Wireless Network Applications, Enterprise Campus Designs, Wireless IST Design, Retail and Manufacturing Design, Small Office/Home Office Design (SOHO)

Text Books:

1. Wireless Security, Merritt Maxim and David Pollino, Osborne/McGraw Hill, New Delhi, 2005
2. Wireless Security Models: Threats and Solutions, Nichols and Lekka, Tata McGraw

Hill, NewDelhi 2006

Reference Books:

1. Behrouz A.Forouzan, —Cryptography & Network Securityll, Tata McGraw Hill, India, New Delhi,2009
2. William Stallings, —Cryptography and Network Security, Prentice Hall, New Delhi, 2006
3. Bruce Schneier, “Applied Cryptography”, John Wiley & Sons, New York, 2004

IV Year – I Semester	L	T	P	C
	3	0	0	3
ETHICAL HACKING (Professional Elective-V)				

Course Objectives:

- The aim of the course is to introduce the methodologies and framework of ethical hacking forenhancing the security.
- The course includes-Impacts of Hacking; Types of Hackers; Information Security Models,Information Security Program, Business Perspective, Planning a Controlled Attack
- Framework of Steps (Reconnaissance, Enumeration, Vulnerability Analysis, Exploitation,Deliverable and Integration)

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

- Explain the concepts related to hacking, ports and protocols, pen testing and virtualization
- Determine the applicable footprinting techniques and scanning methods
- Explain the process of system hacking and Explain the concepts Trojans, backdoors, worms andvirus and it’s countermeasures
- Demonstrate systematic understanding of the concepts of Sniffing and Social Engineering and it’sattacks
- Determine the applicable methods of cryptography, steganography and Vulnerability Assessment

UNIT I:

Introduction to Hacking: Hacking, Types and phases of hacking, **Introduction to Ports & Protocols:** Ports, Protocols, Primary Network Types, **Virtualization & Introduction to Kali Linux:** Virtualization, Virtualization software, supported platforms, **Introduction to Penetration Testing:** Penetration test, Categories and Types of Penetration tests, Structure of Penetration Test Report.

UNIT II:

Footprinting: Footprinting, Types, Using ping and ns Lookup commands in Windows command line, **Scanning:** Scanning, Basics of Scanning, Basic Techniques of Scanning, Enumerating DNS using dns enum, Performing flag scan using hping3.

UNIT III:

Hacking into System: System Hacking, Password Cracking, Default password databases, Manual and Automated Password Cracking, Process of System Hacking, Using Keyloggers, **Trojans & Backdoors:** Trojans, Working of Trojan, Infection Techniques, Attack, Lifecycle and Classification of Virus, Worms, Virus Construction Kit.

UNIT IV:

Sniffing, Packet Analysis & Session Hijacking: Sniffing, Packet Analysis, Types of Sniffing, Active and Passive Sniffing Techniques, Session Hijacking, **Social Engineering:** Social Engineering, Process, Identity Theft, Human and Computer Based Social Engineering Techniques, Phishing Process, Types of Phishing Attacks, Social Engineering Toolkit (SET)

UNIT V:

Cryptography: Cryptography, Digital Signature, Hash Functions, **Steganography:** Steganography Process, watermarking, Steganography Methods and Attacks, Steganography tools, **Vulnerability Assessment:** Vulnerability, The Open Web Application Security Project (OWASP), Prevention, Damn Vulnerable Web Application (DVWA), installation and testing of DVWA

Text Books:

1. Hacking: Be a Hacker with Ethics, Harsh Bothra, Khanna Publications, 2019
2. Ethical Hacking and Penetration Testing Guide, Rafay Baloch, 2014

Reference Books:

1. Kali Linux Wireless Penetration Testing Beginner's Guide, Vivek Ramachandran, CameronBuchanan, Packt Publishing, 2015
2. SQL Injection Attacks and Defense, 1st Edition, Justin Clarke-Salt, Syngress Publication
3. Mastering Modern Web Penetration Testing, Prakhar Prasad, Packt Publishing, October 2016

MOOC'S-NPTL/SWAYAM

IV Year I Semester		L	T	P	C
Computer Graphics Open Elective-III					

Course Objectives:

- To develop, design and implement two and three dimensional graphical structures
- To enable students to acquire knowledge Multimedia compression and animations
- To learn Creation, Management and Transmission of Multimedia objects

Course Outcomes:

After learning the course, the student will be able:

- Illustrate the basics of computer graphics, different graphics systems and applications of computer graphics with various algorithms for line, circle and ellipse drawing objects for 2D transformations
- Apply projections and visible surface detection techniques for display of 3D scene on 2D screen
- Illustrate able to create the general software architecture of programs that use 3D object sets with computer graphics

UNIT I:

Introduction to Graphics: Application areas of Computer Graphics, overview of graphics systems, video-display devices, graphics monitors and work stations and input devices. 2D Primitives: Output primitives- Line, Circle and Ellipse drawing algorithms, Attributes of output primitives, Two dimensional Geometric transformations, Two dimensional viewing Line, Polygon, Curve and Text clipping algorithms.

UNIT II:

3D Concepts: Parallel and Perspective projections, Three dimensional object representation- Polygons, Curved lines, Splines, Quadric Surfaces, Visualization of data sets, 3D transformations, Viewing, Visible surface identification.

UNIT III:

Graphics Programming: Color Models- RGB, YIQ, CMY, HSV, Animations -General Computer Animation, Raster, Keyframe. Graphics programming using OpenGL-Basic graphics primitives, Drawing three dimensional objects, Drawing three dimensional scenes.

UNIT IV:

Rendering: Introduction to shading models, Flat and Smooth shading, Adding texture to faces, Adding shadows of objects, Building a camera in a program, Creating shaded objects

UNIT V:

Overview of Ray Tracing: Intersecting rays with other primitives, Adding Surface texture, Reflections and Transparency, Boolean operations on Objects.

Text Books:

- 1) Donald Hearn, Pauline Baker, Computer Graphics – C Version, second edition, Pearson Education, 2004.
- 2) Schaum's Outline of Computer Graphics Second Edition, Zhigang Xiang, Roy A. Plastock.

Reference Books:

- 1) James D. Foley, Andries Van Dam, Steven K. Feiner, John F. Hughes, Computer Graphics- Principles and practice, Second Edition in C, Pearson Education, 2007.
- 2) F.S. Hill, Computer Graphics using OpenGL, Second edition, Pearson Education, 2003.

E-Resources:

- 1) <http://math.hws.edu/eck/cs424/downloads/graphicsbook-linked.pdf>
- 2) <https://nptel.ac.in/courses/106/106/106106090/>

IVYear I Semester		L	T	P	C
--------------------------	--	----------	----------	----------	----------

Software Testing Methodologies Open Elective-III

Course Objectives:

- To study fundamental concepts in software testing and discuss various software testing issues and solutions in software unit, integration, regression and system testing
- To learn how to plan a test project, design test cases and data, conduct testing, manage software problems and defects, generate a test report
- To expose the advanced software testing concepts such as object-oriented software testing methods, web-based and component-based software testing
- To understand software test automation problems and solutions
- To learn how to write software test documents and communicate with engineers in various forms

Course Outcomes:

By the end of the course, the student should have the ability to:

- Identify and understand various software testing problems, apply software testing knowledge and engineering methods and solve these problems by designing and selecting software test models, criteria, strategies, and methods
- Design and conduct a software test process for a software project
- Analyze the needs of software test automation
- Use various communication methods and skills to communicate with their teammates to conduct their practice-oriented software testing projects
- Basic understanding and knowledge of contemporary issues in software testing, such as component-based, web based and object oriented software testing problems
- Write test cases for given software to test it before delivery to the customer and write test scripts for both desktop and web based applications

UNIT I:

Software Testing: Introduction, Evolution, Myths & Facts, Goals, Psychology, definition, Model for testing, Effective Vs Exhaustive Software Testing.

Software Testing Terminology and Methodology: Software Testing Terminology, Software Testing Life Cycle, Software Testing Methodology.

Verification and Validation: Verification & Validation Activities, Verification, Verification of Requirements, High level and low level designs, verifying code, Validation.

UNIT II:

Dynamic Testing-Black Box testing techniques: Boundary Value Analysis, Equivalence class Testing, State Table based testing, Decision table based testing, Cause-Effect Graphing based testing, Error guessing.

White-Box Testing: need, Logic Coverage criteria, Basis Path testing, Graph matrices, Loop testing, data flow testing, mutation testing.

UNIT III:

Static Testing: Inspections, Structured Walkthroughs, Technical Reviews.

Validation activities: Unit testing, Integration Testing, Function testing, system testing, acceptance testing.

Regression testing: Progressive Vs regressive testing, Regression test ability, Objectives of regression testing, Regression testing types, Regression testing techniques.

UNIT IV:

Efficient Test Suite Management: growing nature of test suite, Minimizing the test suite and its benefits, test suite prioritization, Types of test case prioritization, prioritization techniques, measuring the effectiveness of a prioritized test suite Software Quality Management: Software Quality metrics, SQA models.

Debugging: process, techniques, correcting bugs.

UNIT V:

Automation and Testing Tools: need for automation, categorization of testing tools, selection of testing tools, Cost incurred, Guidelines for automated testing, overview of some commercial testing tools such as Win Runner, Load Runner, Jmeter and JUnit . Test Automation using Selenium tool.

Testing Object Oriented Software: basics, Object oriented testing Testing Web based Systems: Challenges in testing for web based software, quality aspects, web engineering, testing of web based systems, Testing mobile systems.

Text Books:

- 1) Software Testing, Principles and Practices, Naresh Chauhan, Oxford
- 2) Software Testing, Yogesh Singh, CAMBRIDGE

Reference Books:

- 1) Foundations of Software testing, Aditya P Mathur, 2ed, Pearson
- 2) Software testing techniques – Baris Beizer, Dreamtech, second edition.
- 3) Software Testing, Principles, techniques and Tools, M G Limaye, TMH
- 4) Effective Methods for Software testing, Willian E Perry, 3ed, Wiley

E-Resources:

- 1) https://www.tutorialspoint.com/software_testing_dictionary/test_tools.htm

IVYear I Semester		L	T	P	C
-------------------	--	---	---	---	---

Expert System Open Elective-III
--

Course Objectives:

1. Understand the basic techniques of artificial intelligence.
2. Understand the Non-monotonic reasoning and statistical reasoning.

Course Outcomes:

1. Apply the basic techniques of artificial intelligence.
2. Discuss the architecture of an expert system and its tools.
3. Understand the importance of building an expert systems.
4. Understand various problems with an expert systems.

UNIT - I

Introduction to AI programming languages, Blind search strategies, Breadth-first – Depth-first – Heuristic search techniques Hill Climbing – Best first – A Algorithms AO* algorithm – game tress, Min-max algorithms, game playing – Alpha-beta pruning.

UNIT - II

Knowledge representation issues predicate logic – logic programming Semantic nets- frames and inheritance, constraint propagation; Representing Knowledge using rules, Rules-based deduction systems.

UNIT – III:

Introduction to Expert Systems, Architecture of expert systems, Representation and organization of knowledge, Basics characteristics, and types of problems handled by expert systems.

UNIT – IV:

Expert System Tools: Techniques of knowledge representations in expert systems, knowledge engineering, system-building aids, support facilities, stages in the development of expert systems.

UNIT - V

Building an Expert System: Expert system development, Selection of the tool, Acquiring Knowledge, Building process.

Problems with Expert Systems: Difficulties, common pitfalls in planning, dealing with domain experts, difficulties during development.

TEXT BOOKS:

1. Elain Rich and Kevin Knight, “Artificial Intelligence”, Tata McGraw-Hill, New Delhi.
2. Waterman D.A., “A Guide to Expert Systems”, Addison Wesley Longman.

REFERENCE BOOKS:

1. Stuart Russel and other Peter Norvig, “Artificial Intelligence – A Modern Approach”, Prentice-Hall.
2. Patrick Henry Winston, “Artificial Intelligence”, Addison Wesley.
3. Patterson, Artificial Intelligence & Expert System, Prentice Hall India, 1999.
4. Hayes-Roth, Lenat, and Waterman: Building Expert Systems, Addison Wesley.
5. Weiss S.M. and Kulikowski C.A., “A Practical Guide to Designing Expert Systems”, Rowman & Allanheld, New Jersey.

Course Outcomes:

- Describe RPA, where it can be applied and how it's implemented.
- Describe the different types of variables, Control Flow and data manipulation techniques.
- Identify and understand Image, Text and Data Tables Automation.
- Describe how to handle the User Events and various types of Exceptions and strategies.
- Understand the Deployment of the Robot and to maintain the connection.

UNIT I:

Introduction to Robotic Process Automation: Scope and techniques of automation, Robotic process automation, What can RPA do, Benefits of RPA, Components of RPA, RPA platforms, The future of automation.

RPA Basics: History of Automation, What is RPA, RPA vs Automation, Processes & Flowcharts, Programming Constructs in RPA, What Processes can be Automated, Types of Bots, Workloads which can be automated, RPA Advanced Concepts, Standardization of processes, RPA Development methodologies, Difference from SDLC, Robotic control flow architecture, RPA business case, RPA Team, Process Design Document/Solution Design Document, Industries best suited for RPA, Risks & Challenges with RPA, RPA and emerging ecosystem.

UNIT II:

RPA Tool Introduction and Basics:

Introduction to RPA Tool: The User Interface, Variables, Managing Variables, Naming Best Practices, The Variables Panel, Generic Value Variables, Text Variables, True or False Variables, Number Variables, Array Variables, Date and Time Variables, Data Table Variables, Managing Arguments, Naming Best Practices, The Arguments Panel, Using Arguments, About Imported Namespaces, Importing New Namespaces, Control Flow, Control Flow Introduction, If Else Statements, Loops, Advanced Control Flow, Sequences, Flowcharts, About Control Flow, Control Flow Activities, The Assign Activity, The Delay Activity, The Do While Activity, The If Activity, The Switch Activity, The While Activity, The For Each Activity, The Break Activity, Data Manipulation, Data Manipulation Introduction, Scalar variables, collections and Tables, Text Manipulation, Data Manipulation, Gathering and Assembling Data

UNIT III:

Advanced Automation Concepts & Techniques: Recording Introduction, Basic and Desktop Recording, WebRecording, Input/Output Methods, Screen Scraping, Data Scraping, Scraping advanced techniques, Selectors, Defining and Assessing Selectors, Customization, Debugging, Dynamic Selectors, Partial Selectors, RPA Challenge, Image, Text & Advanced Citrix Automation, Introduction to Image & Text Automation, Image based automation, Keyboard based automation, Information Retrieval, Advanced Citrix Automation challenges, Best Practices, Using tab for Images, Starting Apps, Excel Data Tables & PDF, Data Tables in RPA, Excel and Data Table basics, Data Manipulation in excel, Extracting Data from PDF, Extracting a single piece of data, Anchors, Using anchors in PDF.

UNIT IV:

Handling User Events & Assistant Bots, Exception Handling: What are assistant bots, Monitoring system event triggers, Hotkey trigger, Mouse trigger, System trigger, Monitoring image and element triggers, An example of monitoring email, Example of monitoring a copying event and blocking it, Launching an assistant bot on a keyboard event.

Exception Handling: Debugging and Exception Handling, Debugging Tools, Strategies for solving issues, Catching errors.

UNIT V:

Deploying and Maintaining The Bot: Publishing using publish utility, Creation of Server, UsingServer to control the bots, Creating a provision Robot from the Server, Connecting a Robot to Server, Deploy the Robot to Server, Publishing and managing updates, Managing packages, Uploading packages, Deleting packages

TextBooks:

1. Alok Mani Tripathi, "*Learning Robotic Process Automation*", Packt Publishing, 2018.

ReferenceBooks:

1. Frank Casale, Rebecca Dilla, Heidi Jaynes, Lauren Livingston, "Introduction to Robotic Process Automation: a Primer", Institute of Robotic Process Automation, 1st Edition 2015.
2. Richard Murdoch, "Robotic Process Automation: Guide To Building Software Robots, Automate Repetitive Tasks & Become An RPA Consultant", Independently Published, 1st Edition 2018.
3. Srikanth Merianda, "Robotic Process Automation Tools, Process Automation and their benefits: Understanding RPA and Intelligent Automation", Consulting Opportunity Holdings LLC, 1st Edition 2018.
4. Lim Mei Ying, "Robotic Process Automation with Blue Prism Quick Start Guide: Create software robots and automate business processes", Packt Publishing, 1st Edition 2018.

WebReferences:

1. <https://www.uipath.com/rpa/robotic-process-automation>
<https://www.academy.uipath.com>

VYear I Semester		L	T	P	C
MEANSTACKTECHNOLOGIES-MODULEII-ANGULARJSAND MongoDB (SkillOriented Course)					

Course Objectives:

From the course the student will learn

- Translate user requirements into the overall architecture and implementation of new systems and Manage Project and coordinate with the Client
- Writing optimized front end code HTML and JavaScript
- Monitor the performance of web applications & infrastructure and Troubleshooting web application with a fast and accurate a resolution
- Design and implementation of Robust and Scalable Front End Applications

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

- Enumerate the Basic Concepts of Web & Markup Languages
- Develop web Applications using Scripting Languages & Frameworks
- Make use of Express JS and Node JS frameworks
- Illustrate the uses of web services concepts like restful, react js
- Apply Deployment Techniques & Working with cloud platform

List of Exercises:

1.a	CourseName: AngularJS
	ModuleName: AngularApplicationSetup
	Observethelinkhttp://localhost:4200/welcomeonwhichthemCartapplicationis running. Performthebelowactivitiesounderstandthefeaturesofthe application.
	https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_24049616594198490000_shared?collectionId=lex_20858515543254600000_shared&collectionType=Course
1.b	CourseName: AngularJS
	ModuleName: Componentsand Modules
	Createanewcomponentcalled helloandrenderHelloAngularonthe page
	https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_28217843279641040000_shared?collectionId=lex_20858515543254600000_shared&collectionType=Course
1.c	CourseName: AngularJS
	ModuleName: ElementsofTemplate
	Addaneventtothehellocomponenttemplateandwhenitislicked,itshouldchange thecourseName.
	https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_19226434057992030000_shared?collectionId=lex_20858515543254600000_shared&collectionType=Course

1.d	CourseName: AngularJS
	ModuleName: ChangeDetection
	progressivelybuildingthePoolCarzapplication
	https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_2560981637120771000_shared?collectionId=lex_2085851554325460000_0_shared&collectionType=Course
2.a	CourseName: AngularJS
	ModuleName: StructuralDirectives-ngIf
	Create login form with username and password fields. If the user enters the correct credentials, it should render a "Welcome<<username>>" message otherwise it should render "Invalid Login!!! Please try again..." message
	https://infyspringboard.onwingspan.com/web/en/viewer/web-

	module/lex_auth_0127637402260439042595_shared?collectionId=lex_20858515543254600000_shared&collectionType=Course
2.b	CourseName: AngularJS
	ModuleName: ngFor
	Create a courses array and rendering it in the template using ngFor directive in a list format.
	https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_32795774277593590000_shared?collectionId=lex_20858515543254600000_shared&collectionType=Course
2.c	CourseName: AngularJS
	ModuleName: ngSwitch
	Display the correct option based on the value passed to ngSwitch directive.
	https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_23388127475984175000_shared?collectionId=lex_20858515543254600000_shared&collectionType=Course
2.d	CourseName: AngularJS
	Module Name: CustomStructuralDirective
	Create a custom structural directive called 'repeat' which should repeat the element given a number of times.
	https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_24073319904331424000_shared?collectionId=lex_20858515543254600000_shared&collectionType=Course
3.a	CourseName: AngularJS
	ModuleName: AttributeDirectives- ngStyle
	Apply multiple CSS properties to a paragraph in a component using ngStyle.
	https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_24037156998765367000_shared?collectionId=lex_20858515543254600000_shared&collectionType=Course
3.b	CourseName: AngularJS
	ModuleName: ngClass
	Apply multiple CSS classes to the text using ngClass directive.
	https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_3459610297074182000_shared?collectionId=lex_20858515543254600000_shared&collectionType=Course
3.c	CourseName: AngularJS
	ModuleName: CustomAttributeDirective

	<p>Create an attributed directive called 'showMessage' which should display the given message in a paragraph when a user clicks on it and should change the text color to red.</p> <p>https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_14783742359773809000_shared?collectionId=lex_20858515543254600000_shared&collectionType=Course</p>
4.a	CourseName: AngularJS
	ModuleName: PropertyBinding
	Binding image with class property using property binding.
	https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_8951964709153619000_shared?collectionId=lex_20858515543254600000_s_hared&collectionType=Course

INDUSTRIAL/ INTERNSHIP 2