ACADEMIC REGULATIONS (C22)
COURSE STRUCTURE
AND
DETAILED SYLLABUS

DEPARTMENT OF CSE- ARTIFICIAL INTELLIGENCE

FOR

B. TECH FOUR YEAR DEGREE COURSE

(Applicable for the batches admitted from 2022-2023)

CiTY
CHALAPATHI
Institute of Technology
AN AUTONOMOUS INSTITUTE

Approved by AICTE, New Delhi and Permanently Affiliated to JNTUK
Accredited by NBA (CSE, ECE, CIV) and NAAC- A Grade
Phone: +91 9505505566, +91 9848133748, www.city.ac.in
Email: chalapathicit@gmail.com, office@city.ac.in, principal@city.ac.in
Abburi Raghavaiah Nagar, Mothadaka, Guntur,
Andhra Pradesh-522 016

CHALAPATHI INSTITUTE OF TECHNOLOGY

(Autonomous)

DEPARTMENT OF CSE- ARTIFICIAL INTELLIGENCE
COURSE STRUCTURE

I Year–I Semester

S. No.	Course Code	Course Category	Subjects	L	T	P	Credits
1	C221101	HS	Communicative English	3	0	0	3
2	C221102	BS	Calculus for Engineers	3	0	0	3
3	C221104	BS	Applied Chemistry	3	0	0	3
4	C221105	ES	Programming through C	3	0	0	3
5	C221112	ES	Computer Engineering Workshop	1	0	3	3
6	C221108	HS	Communicative English Lab	0	0	3	1.5
7	C221110	BS	Applied Chemistry Lab	0	0	3	1.5
8	C221111	ES	Programming through C Lab	0	0	3	1.5
			Total Credits				**19.5**
9		MC	Induction Program				--

Category	Credits
Basic Science Courses	3+3+1.5 = 7.5
Engineering Science Courses	3+3+1.5 = 7.5
Humanities and Social Science Courses	3+1.5=4.5
TOTAL CREDITS	19.5

I Year –II Semester

S. No.	Course Code	Course Category	Subjects	L	T	P	Credits
1	C221201	ES	Data Structures	3	0	0	3
2	C221202	BS	Linear Algebra & Multi-Variable Calculus	3	0	0	3
3	C221203	BS	Applied Physics	3	0	0	3
4	C221205	ES	Python Programming	3	0	0	3
5	C221208	ES	Digital Logic Design	3	0	0	3
6	C221211	ES	Data Structures Lab	0	0	3	1.5
7	C221212	BS	Applied Physics Lab	0	0	3	1.5
8	C221214	ES	Python Programming Lab	0	0	3	1.5
			Total Credits				**19.5**
9		MC	Technology and Society	1	0	0	-

Category	Credits
Basic Science Courses	3+3+1.5 = 7.5
Engineering Science Courses	3+3+3+1.5+1.5 = 12
TOTAL CREDITS	19.5

CHALAPATHI INSTITUTE OF TECHNOLOGY

(Autonomous)

DEPARTMENT OF CSE- ARTIFICIAL INTELLIGENCE
COURSE STRUCTURE

II Year–I Semester

S. No	Course Code	Course Category	Subjects	L	T	P	Credits
1	C2221011	BS	Transforms & Partial Differential Equations	3	0	0	3
2	C2221052	PC	Mathematical Foundations of Computer Science	3	0	0	3
3	C2221051	PC	Object Oriented Programming through JAVA	3	0	0	3
4	C2221431	PC	Computer Organization	3	0	0	3
5	C2221432	PC	Database Management Systems	3	0	0	3
6	C2221433	PCL	Object Oriented Programming through JAVA Lab	0	0	3	1.5
7	C2221055	PCL	Computer Organization Lab	0	0	3	1.5
8	C2221434	PCL	Database Management Systems Lab	0	0	3	1.5
9	C2221435	SC	Skill Oriented Course-I* (Applications of Python-NumPy)	1	0	2	2
			Total Credits				**21.5**
10		MC	Essence of Indian Traditional Knowledge	2	0	0	0

Category	Credits
Basic Science Courses	3
Professional Core Courses	3+3+3+3+1.5+1.5+1.5 = 16.5
Skill Oriented Course	2
TOTAL CREDITS	21.5

II Year –II Semester

S. No	Course Code	Course Category	Subjects	L	T	P	Credits
1	C2222011	BS	Probability and Statistics	3	1	0	3
2	C2222432	PC	Design & Analysis of Algorithms	3	1	0	3
3	C2222433	PC	Data Warehousing and Mining	3	0	0	3
4	C2222053	PC	Formal Languages and Automata Theory	3	1	0	3
5	C2222431	PC	Introduction to Artificial Intelligence	3	1	0	3
6	C2222435	PCL	Data Mining using Python Lab	0	0	3	1.5
7	C2222434	PCL	Introduction to Artificial Intelligence Lab	0	0	3	1.5
8	C2222436	PCL	Web Application Development Lab	0	0	3	1.5
9	C2222047	SC	Skill Oriented Course-II* (Applications of Python-Pandas)	1	0	2	2
			Total Credits				**21.5**
10	C222201M	MC	Industrial/Research Internship	0	0	3	0

Category	Credits
Basic Science Courses	3
Professional Core Courses	3+3+3+3+1.5+1.5+1.5 = 16.5
Skill Oriented Course	2
TOTAL CREDITS	21.5

I Year – I Semester	COMMUNICATIVE ENGLISH	L	T	P	C
		3	0	0	3

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
- Focus on appropriate reading strategies for comprehension of various academic texts and authentic materials spoken by native speakers
- 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

Course Outcomes:

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

- Explain the importance of Paragraph Writing on sign posts
- Illustrate Nehru's letter to his daughter Indira on her birthday
- Convince Stephen Hawking theory with own words.
- Identify the influence of Wangari Mathai in Green belt movement
- Develop the story on Stay Hungry Stay Foolish

UNIT - I

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.
Pronunciation: Vowels, Consonants, Plural markers and their realizations

UNIT - II

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.
Pronunciation: Past tense markers, word stress-di-syllabic words

UNIT - III

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.
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.
Pronunciation: word stress-poly-syllabic words.

UNIT - IV

Lesson-1: Like 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.
Reading for Writing: Information transfer; describe, compare, contrast, identify significance/trends based on information provided in figures/charts/graphs/tables. Writing SOP, writing for media.
Vocabulary: Technical vocabulary from across technical branches (20 words) GRE Vocabulary (20 words) (Antonyms and Synonyms, Word applications) Cloze Encounters.
Grammar: Quantifying expressions - adjectives and adverbs; comparing and contrasting; degrees of comparison; use of antonyms

Pronunciation: Contrastive Stress

UNIT - V:

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.
Reading for Writing: Writing academic proposals- writing research articles: format and style.
Vocabulary: Technical vocabulary from across technical branches (20 words) GRE Vocabulary (20 words) (Antonyms and Synonyms, Word applications) Coherence, matching emotions.
Grammar: Editing short texts – identifying and correcting common errors in grammar and usage (articles, prepositions, tenses, subject verb agreement)
Pronunciation: Stress in compound words

Text Books:

1. **"Infotech English**", Maruthi Publications, Second Edition,2021 (Detailed)
2. "**The Individual Society**", Pearson Publications,2019 (Non-detailed)

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, 201

E-Resources and Other Digital Material:

1. https://drive.google.com/drive/folders/1sdGU19sjG-Vft0_ZGFJp6ZIA-IkJU8qJ?usp=sharing
2. https://drive.google.com/drive/folders/1fcDZDJEge4Y9cZh2flO_Oea6AZWwmyL_?usp=sharing
3. https://drive.google.com/drive/folders/1t2JD9po252PHI80O1vHNJ0hAy1k1C46C?usp=sharing

I Year- **I Semester**	**CALCULUS FOR ENGINEERS** (Common to all branches)	**L**	**T**	**P**	**C**
		3	**0**	**0**	**3**

Course Objectives:

- To enlighten the learners in the concept of differential equations.
- To equip the students with standard concepts and tools at an intermediate to advanced level mathematics to develop the confidence and ability among the students to handle various real-world problems and their applications.

Course Outcomes:

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

- Utilize mean value theorem store al life problems(L3)
- Solve the differential equations related to various engineering fields(L3)
- Determine the solutions for differential equations for circuit analysis (L3)
- Interpretthephysicalmeaningofdifferentoperatorssuchasgradient,curlanddivergence(L5)
- Estimate the work done against a field , circulation and flux using vector calculus(L5)

UNIT–I:

Differential equations of first order and first degree: (10hrs)

Lineardifferentialequations–Bernoulli'sequations–Exactequationsandequationsreducibletoexactform.

Applications: Newton'sLawofcooling–Lawofnaturalgrowthanddecay–Orthogonaltrajectories–Electricalcircuits.

UNIT– II:

Linear differential equations of higher order: (10hrs)

Homogeneous and Non-homogeneous differential equations of higher order with constant coefficients–with non-homogeneous term of the typee^{ax}, sinax, cosax, polynomialsinxn, eaxV(x) andxnV(x)–MethodofVariationofparameters,

Applications:LCRcircuit.

UNIT–III:

Mean value theorems:

(10hrs)

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.

UNIT IV:

Vector Differentiation: (8 hrs)

Vector Differentiation: Gradient – Directional derivative – Divergence – Curl –Scalar Potential.

UNIT V:

Vector Integration: **(10 hrs)**

Vector Integration: Line integral – Work done – Area – Surface and volume integrals – Vector integral theorems: Greens, Stokes and Gauss Divergence theorems (without proof).

Text Books:

1. B.S.Grewal, Higher Engineering Mathematics, 44thEdition, Khanna Publishers.
2. B.V.Ramana,Higher Engineering Mathematics ,2007Edition ,Tata McGraw-Hill Education.

Reference Books:

1. ErwinKreyszig,AdvancedEngineeringMathematics,10thEdition,Wiley-India.
2. JoelHass,ChristopherHeilandMauriceD.Weir, Thomascalculus, 14thEdition,Pearson.
3. LawrenceTuryn,AdvancedEngineeringMathematics,CRCPress,2013.
4. SrimanthaPal,SCBhunia,EngineeringMathematics, OxfordUniversityPress.

E-Resources:

1. https://archive.nptel.ac.in/courses/111/106/111106100/
2. https://www.digimat.in/nptel/courses/video/111108081/L01.html

I Year– I Semester	APPLIED CHEMISTRY (Common to all branches)	L	T	P	C
		3	0	0	3

Knowledge of basic concepts of Chemistry for Engineering students will help them as professional engineers later in design and material selection, as well as utilizing the available resources.

COURSE OBJECTIVES

- Importance of usage of plastics in household appliances and composites in aerospace and automotive industries.
- Outline the basics for the construction of electrochemical cells, batteries and fuel cells. Understand the mechanism of corrosion and how it can be prevented.
- Explain the preparation of semiconductors and nanomaterials, engineering applications of nanomaterials, superconductors andliquid crystals.
- Outline the basics of computational chemistry and molecularswitches
- Recall the increase in demand for power and hence alternative sources of power are to be studied. Advanced instrumental techniques are introduced.

Course Outcomes:

- Analyse the different types of composite plastic materials and use of plastic in design would be understood.
- 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.
- Synthesize nanomaterials for modern advances of engineering technology and summarize the preparation of semiconductors. Analyse the applications of liquid crystals and super conductors.
- Obtain the knowledge of computational chemistry and molecular machines
- Analyze the principles of different analytical instruments and their applications. Design models for energy by different natural sources.

UNIT I: POLYMER TECHNOLOGY 12 hrs

Part-I Polymerisation: - Introduction, types of polymerisation-Addition and condensation, free radical mechanisam, methods of polymerization (emulsion and suspension).

Plastics: Types-Thermo plastics, Thermo sets, compounding of plastics, plastic moulding methods (compression, injection, blown film and extrusion), preparation, properties and applications of some plastic compounds (PVC, Teflon and Bakelite), recycling of e-plastic waste.

Part-II Elastomers :- Natural rubber, vulcanization, preparation, properties and applications of some synthetic rubbers (Buna S, Thiokol).

Composite materials: Conducting polymers, biodegradable polymers.

UNIT II: ELECTROCHEMICAL CELLS AND CORROSION 10 hrs

Part-I Electro chemical cells:-Single electrode potential, EMF calculation, electrochemical series and uses, standard electrodes (Hydrogen electrode, calomel electrode), Batteries (Dry cell, Ni-Cd cells), fuel cells (H2-O2, CH3OH-O2).

Part-II ***Corrosion:-***Definition, theories of corrosion (chemical and electrochemical), Types of corrosion (Galvanic corrosion, differential aeration corrosion, stress corrosion), factors influencing rate of corrosion (Nature of metal, Nature of environment), corrosion control methods (Cathodic protection - (Sacrificial, impressed methods),Protective coatings-(Hot dipping, Metallic coatings, electroplating and electro less plating of nickel))

UNIT III: MATERIAL CHEMISTRY 8 hrs

Part I: Non-elemental semiconducting materials: - Stoichiometric, chalcogen photo/semiconductors, preparation of semiconductors (Distillation, Zone refining, Czochralski crystal pulling, Epitaxy, Diffusion, Ion implantation),P-N junction diode.

Nano materials:- Introduction, sol-gel method, Types, preparation and applications of fullerenes, carbon nanotubes.

Part II:

Liquid crystals:- Introduction-types-applications.

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

UNIT IV: ADVANCED CONCEPTS/TOPICS IN CHEMISTRY 8 hrs

Computational chemistry: Introduction to computational chemistry. **Molecular motors and machines**: Introduction, characteristics of molecular motors, synthesis of Rotaxanes and Catenanes, linear motions in rotaxanes, an autonomous light-powered molecular motor.

UNIT V: SPECTROSCOPIC TECHNIQUES & NON-CONVENTIONAL ENERGY SOURCES 12 hrs

Part I: **SPECTROSCOPIC TECHNIQUES**

Introduction of Electromagnetic spectrum, UV – (laws of absorption, instrumentation of UV, chromophores and auxochromes, intensity shifts, applications), FT-IR Introduction, instrumentation and applications.

Part II: - NON-CONVENTIONAL ENERGY SOURCES

Design, working, schematic diagram, advantages and disadvantages of photovoltaic cell, hydropower, geothermal power, tidal and wave power, ocean thermal energy conversion.

Standard 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 Co. (Latest edition).

Reference:

1. K. Sesha Maheshwaramma and Mridula Chugh, “**Engineering Chemistry**”, Pearson India Edn.
2. O.G. Palana, “**Engineering Chemistry**”, Tata McGraw Hill Education Private Limited, (2009).
3. Y.Bharathi kumari “**Textbook of Applied chemistry”** , JNTU Kedition.
4. B. S. Murthy, P. Shankar and others, “**Textbook of Nanoscience and Nanotechnology**”, University press (latest edition)

E-Resources:

1. https://archive.nptel.ac.in/courses/104/105/104105039/ Introduction to Polymers
2. https://onlinecourses.nptel.ac.in/noc22_mm17/preview about corrosion
3. http://nptel.iitm.ac.in introduction to UV spectro scopy

I Year – I Semester	PROGRAMMING THROUGH C (Common to all branches)	L	T	P	C
		3	0	0	3

Course Objectives:

The objectives of Programming for Problem Solving Using Care

1. To learn about the computer systems, computing environments, develop in to a computer program and Structure of a C Program
2. To gain knowledge of the operators, selection, control statements and repetition in C
3. To learn about the design concept so far rays, strings, enumerated structure and union types and their usage.
4. To assimilate about pointers, dynamic memory allocation and know the significance of Pre processor.
5. To assimilate about File I/O and significance of functions

Course Outcomes:

Upon the completion of the course the student will learn

1. To write algorithms and- to draw flow charts for solving problems
2. To convert flowcharts/algorithms to C Programs, compile and debug programs
3. To use different operators, data types and write programs that use two-way/multi-way selection
4. To select the best loop construct for a given problem
5. To design and implement programs to analyze the different pointer applications
6. To decompose a problem into functions and to develop modular reusable code
7. To apply File I/O operations

UNIT I:

Introduction to Computers: Creating and running Programs, Algorithm, Flow charts, Structure of C program.

Introduction to the C Language: Background, C Programs, Identifiers, Data Types,- Variable, Constants, Input/output, Programming Examples.

UNIT II:

Operators: Expressions Precedence and Associativity, Evaluating Expressions, Type Conversion Statements, Simple Programs. Selection & Making Decisions: Logical Data and Operators, Two Way Selection, Multiway Selection, More Standard Functions.

Repetition: Concept of Loop, Pretest and Post-test Loops, Initialization and Updating, Event and Counter Controlled Loops, Loops in C, Other Statements Related to Looping.

UNIT III:
Arrays: Concepts, Using Array in C, Array Application, Two Dimensional Arrays, Multi-dimensional Arrays, Programming Example.

Strings: String Concepts, C String, String Input/ Output Functions, Arrays of Strings, String Manipulation Functions.

Enumerated, Structure and Union: The Type Definition (Typedef), Enumerated Types, Structure, Unions and Programming Application.

UNIT IV:
Pointers: Introduction, Pointers to pointers, Compatibility, scope and storage classes.
Pointer Applications: Arrays and Pointers, Pointer Arithmetic and Arrays, Memory Allocation Function, Array of Pointers, Programming Application.

UNIT V:
Functions: Designing, Structured Programs, Function in C, User Defined Functions, Inter-Function Communication, Standard Functions, Passing Array to Functions, Passing Pointers to Functions, Recursion

Text Input / Output: Files, Streams, Standard Library Input / Output Functions, Formatting Input /Output Functions, Character Input/ Output Functions

Text Books:

1. Programming for Problem Solving, BehrouzA. Forouzan, RichardF. Gilberg, CENGAGE.
2. The C Programming Language, Brian W.Kernighan, Dennis M. Ritchie, 2e,Pearson.

Reference Books:

1. Computer Fundamentals and Programming, Sumithabha Das, McGrawHill.
2. Programming in C, Ashok N.Kamthane, Amit Kamthane, Pearson.
3. Computer Fundamentals and Programming in C, PradipDey, ManasGhosh, OXFORD.

I Year–I Semester	COMPUTER ENGINEERING WORKSHOP	L	T	P	C
		1	0	3	3

Course Objectives:

The objective of this course is to

1. Explain the internal parts of a computer, peripherals, I/O ports, connecting cables
2. Demonstrate basic command line interface commands on Linux
3. Teach the usage of Internet for productivity and self paced life long learning
4. Describe about Compression, Multimedia and Antivirus tools
5. Demonstrate Office Tools such as Word processors, Spread sheets and Presentation tools

Course Outcomes:

Student should be able to:

1. Assemble and disassemble component of a PC
2. Construct a fully functional virtual machine, Summarize various Linux operating system commands,
3. Recognize characters & extract text from scanned images, Create audio files and podcasts
4. Develop and preparation of a Simple Website /Home page using HTML tags and CSS.
5. Demonstration and Practice on Text Editors.

Computer Hardware:

Experiment 1: Identification of peripherals of a PC, Laptop, Server and Smart Phones: Prepare a report containing the block diagram along with the configuration of each component and its functionality, Input/ Output devices, I/O ports and interfaces, main memory, cache memory and secondary storage technologies, digital storage basics, networking components and speeds.

Operating Systems:

Experiment2: Windows installation setup:

Setting up and configuring a new Windows installation

Setting up and configuring an existing Windows installation

Exporting and packaging an existing Windows installation into a portable format

Experiment3: Linux installation setup:

Setting up and configuring a new Linux installation

Setting up and configuring an existing Linux installation

Exporting and packaging an existing Linux installation into a portable format

Experiment4: Operating System installation:
Installing an Operating System such as Linux on Computer hardware

Experiment5: Linux Operating System commands:
General command syntax
Basic help commands
Basic File system commands
Date and Time
Basic Filters and Text processing
Basic File compression commands
Miscellaneous: apt-get, vi editor
Networking and Internet:
Experiment6: **Networking Commands:**
ping, ssh, ifconfig, scp, netstat, ipstat, nslookup, traceroute, telnet, host, ftp, arp, wget,route
Experiment7: **Internet Services:**
Web Browser usage and advanced settings like LAN, proxy, content, privacy, security, cookies, extensions/plugins
Antivirus installation, configuring a firewall, blocking pop-ups
Email creation and usage, Creating a Digital Profile on LinkedIn
Office Tools:
Experiment8: Demonstration and Practice on Text Editors like Notepad++ , Sublime Text, Atom ,Brackets ,Visual code,etc
Experiment 9: Demonstration and practice on Microsoft Word
Experiment 10: Demonstration and practice on Power Point
Experiment 11: Demonstration and practice on Microsoft Excel.

Text Books:

1. Computer Fundamentals, AnitaGoel, Pearson Education,2017
2. PC Hardware Trouble Shooting Made Easy, TMH

References Books:

- Essential Computer and IT Fundamentals for Engineering and Science Students, Dr.N.B.Vekateswarlu, S.Chand

E-Resources:

1) https://explorersposts.grc.nasa.gov/post631/2006-2007/computer _basics/ ComputerPorts.doc

I Year – I Semester	COMMUNICATIVE ENGLISH LAB (Common to all branches)	L	T	P	C
		0	0	3	1.5

COURSE OUTCOMES:

1. Develop active and authentic listening comprehension skills relevant for the professional world.
2. Execute web related (On-line) communication with felicity of expression.
3. Apply relevant speech patterns including standard pronunciation.
4. Demonstrate Proficiency in Interpersonal Communication with fluency and accuracy

UNIT I:

Vowels, Consonants, Pronunciation, Phonetic Transcription, Common Errors in Pronunciation,

UNIT II:

Word stress-di-syllabic words, poly-syllabic words, weak and strong forms, contrastive stress (Homographs

UNIT III:

Stress in compound words, rhythm, intonation, accent neutralization.

UNIT IV:

Listening to short audio texts and identifying the context and specific pieces of information to answer a series of questions in speaking.

UNIT V:

Newspapers reading; Understanding and identifying key terms and structures useful for writing reports.

Text Book:

1. "Infotech English", Maruthi Publications.

Reference Books:

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 Hewing, Cambridge University Press.
5. English Pronunciation Dictionary- Daniel Jones, Cambridge University Press.
6. English Phonetics for Indian Students- P. Bala Subramanian, Mac Millan Publications.

E-RESOURCES AND OTHER DIGITAL MATERIAL:

1. https://drive.google.com/file/d/15N3ZHBh0SuV0DuQlcA3TMsavY-3ItNAe/view
2. https://archive.nptel.ac.in/content//storage2/109/103/109103183/MP4/mod01lec01.mp4
3. https://archive.nptel.ac.in/content//storage2/109/103/109103183/MP4/mod01lec02.mp4
4. https://archive.nptel.ac.in/content//storage2/109/103/109103183/MP4/mod01lec03.mp4
5. English Language Lab Software by K-VAN Solutions Pvt. Ltd.

I Year – I Semester	APPLIED CHEMISTRY LAB	L	T	P	C
		0	0	3	1.5

Course Outcomes: The students entering into the professional course have practically very little exposure to lab classes. The experiments introduce volumetric analysis; redox titrations with different indicators; EDTA titrations; then they are exposed to a few instrumental methods of chemical analysis. Thus at the end of the lab course, the student is exposed to different methods of chemical analysis and use of some commonly employed instruments. They thus acquire some experimental skills.

1. Introduction to Chemistry laboratory – Molarity, Normality, primary, secondary standard solutions, volumetric titrations, quantitative analysis.
2. Determination of HCl using standard Na2CO3 solution.
3. Determination of alkalinity of a sample containing Na2CO3 and NaOH.
4. Determination of Mn+2 using standard oxalic acid solution.
5. Determination of total hardness of water using standard EDTA solution.
6. Determination of Zinc by using standard EDTA solution.
7. Determination of ferrous iron using standard K2Cr2O7 solution.
8. Determination of Cu+2 using standard hypo solution.
9. Determination of phosphoric content in soft drinks.
10. Determination of the P^H by PH- metry method.
11. Determination of the concentration of strong acid vs strong base (by conductometric method).
12. Determination of the concentration of strong acid vs weak base (by conductometric method).
13. Determination of strong acid vs strong base (by potentiometric method).
14. Estimation of Vitamin C content in lemon juice/tablets.
15. Preparation of Bakelite (demonstration only).
16. Determination of Mg+2 present in an antacid.

Of the above experiments at-least 10 assessment experiments shouldbe completed in a semester.

Reference Books:

1. A Textbook of Quantitative Analysis, Arthur J. Vogel.
2. A Textbook of Applied chemistry, by Y. Bharathi Kumari

I Year - I Semester	PROGRAMMING THROUGH C LAB (Common to all branches)	L	T	P	C
		0	0	3	1.5

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

- Express knowledge on various concepts of a C language.
- Sketch flowcharts and write algorithms.
- Design and development of C problem solving skills.
- Describe and develop modular programming skills
- Test and debug a program

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 7 inches and width of 5 inches.
3. Write a C program to display multiple variables.

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

Exercise 11:

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 12:

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

Exercise 13:

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 get the largest element of an array using the function.

Exercise 14:

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 remove a file from the disk.

I Year - II Semester	**DATA STRUCTURES (Common to all branches)**	**L**	**T**	**P**	**C**
		3	**0**	**0**	**3**

Course Objectives:
The objective of the course is to

- Introduce the fundamental concept of data structures and abstract data types Emphasize the importance of data structures in developing and implementing efficient algorithms
- Describe how arrays, records, linked structures, stacks, queues, trees, and graphs are represented in memory and used by algorithms

Course Outcomes:
After completing this course a student will be able to:

- Summarize the properties, interfaces, and behaviors of basic abstract data types
- Discuss the computational efficiency of the principal algorithms for sorting & searching
- Use arrays, records, linked structures, stacks, queues, trees, and Graphs in writing programs
- Demonstrate different methods for traversing trees
- Evaluate different applications by using Graphs

UNIT I
Data Structures - Definition, Classification of Data Structures, Operations on Data Structures, Abstract Data Type (ADT), Preliminaries of algorithms. Time and Space complexity.
Searching - Linear search, Binary search, Fibonacci search.
Sorting- Insertion sort, Selection sort, Exchange (Bubble sort, quick sort), distribution (radix sort), merging (Merge sort) algorithms.

UNIT III
Stacks: Introduction to Stacks, Array Representation of Stacks, Operations on Stacks, Linked list Representation of Stacks, Operations on Linked Stack, Applications-Reversing list, Infix to Postfix Conversion, Evaluating Postfix Expressions.
Queues: Introduction to Queues, Representation of Queues-using Arrays and using Linked list, Implementation of Queues-using Arrays and using Linked list, Application of Queues- Circular Queues.

UNIT III
Linked List: Introduction, Operations on Single, Reversing Single Linked list, Applications on Single Linked list- Polynomial Expression Representation, Addition and Multiplication, Sparse Matrix Representation using Linked List, Advantages and Disadvantages of Single Linked list, Double Linked list-Insertion, Deletion, Circular Linked list-Insertion, Deletion.

UNIT IV

Trees: Basic Terminology in Trees, Binary Trees-Properties, Representation of Binary Trees using Arrays and Linked lists. Binary Search Trees- Basic Concepts, BST Operations: Insertion, Deletion, Tree Traversals, Applications-Expression Trees, Heap Sort, Balanced Binary Trees- AVL Trees, Insertion, Deletion and Rotations.

UNIT V

Graphs: Basic Concepts, Representations of Graphs-Adjacency Matrix and using Linked list, Graph Traversals (BFT & DFT), Applications- Minimum Spanning Tree Using Prims & Kruskals Algorithm, Dijkstra's shortest path, Transitive closure, Warshall's Algorithm.

Text Books:

1. Data Structures Using C. 2nd Edition.Reema Thareja, Oxford.
2. Data Structures and algorithm analysis in C, 2nded, Mark Allen Weiss.

Reference Books:

1. Fundamentals of Data Structures in C, 2nd Edition, Horowitz, Sahni, Universities Press.
2. Data Structures: A PseudoCode Approach, 2/e, Richard F.Gilberg, Behrouz A. Forouzon, Cengage.
3. Data Structures with C, Seymour Lipschutz TMH

e-Resources:

- http://algs4.cs.princeton.edu/home/
- https://faculty.washington.edu/jstraub/dsa/Master_2_7a.pdf

I Year – II Semester	LINEAR ALGEBRA AND MULTI-VARIABLE CALCULUS (Common to all branches)	L	T	P	C
		3	0	0	3

Course Objectives:

- To instruct the concept of Matrices in solving linear algebraic equations
- To elucidate the different numerical methods to solve non linear algebraic equations
- Todisseminatetheuseofdifferentnumericaltechniquesforcarryingoutnumericalintegration.
- To equip the students with standard concepts and tools at an intermediate to advanced level mathematics to develop the confidence and ability among the students to handle various real world problems and their applications.

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

- Solve system of linear algebraic equations using Gausse limination , Gauss Jordan ,Gauss Seidel(L3)
- Evaluate the approximate roots of polynomial and transcendental equations by different algorithms (L5)
- Apply Newton's forward & backward interpolation and Lagrange's formulae for equal and unequal intervals (L3)
- applynumericalintegraltechniquestodifferentEngineeringproblems(L3)
- applydifferentalgorithmsforapproximatingthesolutionsofordinarydifferentialequationswithinitialconditionstoits analytical computations(L3)

UNIT–I:

Solving systems of linear equations , Eigen values and Eigen vectors: (10hrs)

Rank of a matrix by echelon form and normal form – Solving system of homogeneous and non-homogeneous linear equations–Gauss Elimination method–Eigen values and Eigen vectors and properties

Unit –II:

Cayley – Hamilton theorem and Quadratic forms: (10hrs)

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 .Singular values of a matrix ,singular value decomposition.

UNIT–III:

Iterative methods: (8hrs)

Introduction–Bisectionmethod–Secantmethod–Methodoffalseposition–Iterationmethod–Newton-Raphsonmethod–JacobiandGauss-Seidelmethodsforsolvingsystemofequations numerically.

UNIT–IV:

Interpolation: (10hrs)

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 – V:

Numerical differentiation and integration, Solution of ordinary differential equations with initial conditions: (10hrs)

Numerical differentiation using interpolating polynomial – Trapezoidal rule– Simpson's 1/3rdand 3/8thrule–Solution of initial value problems by Taylor's series–Picard's method of successive approximations–Euler'smethod–Runge-Kutta method(second and fourth order).

Text Books:

1. B.S.Grewal,HigherEngineeringMathematics,44thEdition,KhannaPublishers.
2. B.V.Ramana,HigherEngineeringMathematics,2007Edition,TataMc.GrawHillEducation.
3. DavidPoole,LinearAlgebra-Amodernintroduction,4thEdition,Cengage.

Reference Books:

1. StevenC.Chapra,AppliedNumericalMethodswithMATLABforEngineeringandScience,Tata Mc.GrawHillEducation.
2. M.K.Jain,S.R.K.IyengarandR.K.Jain,NumericalMethodsforScientificandEngineeringComputation,NewAgeInternationalPublications.
3. LawrenceTuryn,AdvancedEngineeringMathematics,CRCPress.

E-Resources:

1. https://archive.nptel.ac.in/courses/111/106/111106051/
2. https://nptel.ac.in/courses/111106101

I Year– II Semester	APPLIED PHYSICS (Common to all branches)	L	T	P	C
		3	0	0	3

Course Objectives:

- Bridging the gap between the physics in school at 10+2 level and UG level engineering courses.
- To identify the importance of the optical phenomenon i.e. interference, diffraction and polarization related to its Engineering applications
- Understand the mechanism of emission of light, utilization of lasers as coherent light sources for low and high energy applications, study of propagation of light through optical fibers and their implications in optical communications.
- To explain the significant concepts of dielectric and magnetic materials that leads to potential applications in the emerging micro devices.
- Enlightenment of the concepts of Quantum Mechanics and to provide fundamentals of deBroglie matter waves, quantum mechanical wave equation and its application, the importance of free electron theory for metals. Metals- Semiconductors-Insulators concepts utilization of transport phenomenon of charge carriers in semiconductors.
- To understand the physics of Semiconductors and their working mechanism. To give an impetus on the subtle mechanism of superconductors using the concept of BCS theory and their fascinating applications.

Course Outcomes:

- Explain the need of coherent sources and the conditions for sustained interference and its applications. Illustrate the concept of polarization of light.
- Understand various types of emission of radiation and working principles of various types of lasers. Classify optical fibers based on refractive index profile and mode of propagation.
- Summarize various types of polarization of dielectrics. Classify the magnetic materials based on susceptibility and their temperature dependence and explain the applications of magnetic materials.
- Describe the dual nature of matter, significance of wave function. Identify the role of classical and quantum free electron theory in the study of electrical conductivity.
- Outline the properties and applications of charge carriers in semiconductors. Classify superconductors based on Meissner’s effect.

Unit-I: Wave Optics 12hrs

Interference: Principle of superposition –Interference of light - Interference in thin films (Reflection Geometry) & applications - Colors in thin films-Newton's Rings- Determination of wavelength and refractive index.

Diffraction: Introduction - Fresnel and Fraunhofer diffraction - Fraunhofer diffraction due to single slit, double slit - N-slits (Qualitative) – Diffraction Grating - Dispersive power and resolving power of Grating (Qualitative).

Polarization: Introduction-Types of Polarization – Polarization by reflection, refraction and double refraction-Nicol's Prism-Half wave and Quarter wave plates

Unit-II: Lasers and Fiber optics 8hrs

Lasers: Introduction – Characteristics of laser – Spontaneous and Stimulated emissions of radiation – Einstein's coefficients – 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-III: Dielectric and Magnetic Materials 10hrs

Dielectric Materials: Introduction - Dielectric polarization - Dielectric polarizability, Susceptibility and Dielectric constant. Types of Polarizations-Electronic (Quantitative), Ionic (Quantitative) and Orientation Polarizations(Qualitative)-Lorentz internal field- Clausius-Mossotti equation.
Magnetic Materials: Introduction - Magnetic dipole moment - Magnetization-Magnetic susceptibility and permeability - Origin of permanent magnetic moment - Classification of magnetic materials: Dia, para, Ferro, anti-ferro&Ferri magnetic materials - Domain concept for Ferromagnetism & Domain walls (Qualitative) - Hysteresis - soft and hard magnetic materials-Engineering applications.

Unit IV: Quantum Mechanics and Free Electron Theory 8hrs

Quantum Mechanics: Introduction-Matter waves-Dual nature of matter – Heisenberg's Uncertainty Principle – Significance and properties of wave function – Schrodinger's time independent and dependent wave equations– Particle in a one-dimensional infinite potential well and three dimensions.

Free Electron Theory: Classical free electron theory (Qualitative with discussion of merits and demerits) – Quantum free electron theory– Equation for electrical conductivity based on quantum free electron theory-Fermi-Dirac distribution- Density of states (3D) - Fermi energy.

Unit – V: Semiconductors and Superconductors 12hrs

Semiconductors: Introduction- Intrinsic semiconductors – Density of charge carriers – Electrical conductivity – Fermi level –extrinsic semiconductors- density of charge carriers – dependence of fermi energy on carrier concentration and temperature – Drift and diffusion currents-Einstein's equation-Hall effect – Hall coefficient –Applications of Hall effect.
Superconductors: Introduction – Properties of superconductors- Meissner effect – Type I and Type II superconductors- BCS theory (Qualitative)-Josephson effects(AC and DC) - High T_c superconductors – Applications of superconductors.

Text books:

1. M.N.Avadhanulu, P.G.Kshirsagar& TVS Arun Murthy" A Text book of Engineering Physics"- S.Chand Publications, 11th Edition 2019.
2. Engineering Physics" by D.K.Bhattacharya and PoonamTandon, Oxford press (2015).
3. Applied Physics by P.K.Palanisamy SciTech publications.

Reference Books:

1. Fundamentals of Physics – Halliday, Resnick and Walker, John Wiley &Sons
2. Engineering Physics by M.R.Srinivasan, New Age international publishers (2009).
3. Shatendra Sharma, Jyotsna Sharma, "Engineering Physics", Pearson Education, 201
4. Engineering Physics - Sanjay D. Jain, D. Sahasrabudhe and Girish, University Press
5. Semiconductor physics and devices- Basic principle – Donald A, Neamen, McGraw Hill
6. B.K. Pandey and S. Chaturvedi, Engineering Physics, Cengage Learning

E-Resources:

1. https://archive.nptel.ac.in/courses/122/107/122107035/ Introduction to interference and diffraction
2. https://archive.nptel.ac.in/courses/115/102/115102124/ About lasers
3. https://archive.nptel.ac.in/courses/115/101/115101107/ Introduction to Quantum mechanics

I Year – II Semester	PYTHON PROGRAMMING (Common to all branches)	L	T	P	C
		3	0	0	3

Course Outcomes

- Describe the essential programming skills in computer programming concepts like data types, containers
- Solve coding tasks related conditional execution, loops
- Experiment with various Data structures in interpreted Language and to build modules and packages for real software needs
- Understand the basic concepts of object oriented programming
- Develop python programs with appropriate Exception handling

UNIT I

Introduction: Introduction to Python, Program Development Cycle, Input, Processing, and Output, Displaying Output with the Print Function, Comments, Variables, Reading Input from the Keyboard, Performing Calculations, Operators. Type conversions, Expressions, More about Data Output.

Data Types, and Expression: Strings Assignment, and Comment, Numeric Data Types and Character Sets, Using functions and Modules.

UNIT II

Decision Structures and Boolean Logic and Control Statement: if, if-else, if-elif-else Statements, Nested Decision Structures, Comparing Strings, Logical Operators, Boolean Variables.
Repetition Structures: Introduction, while loop, for loop, Calculating a Running Total, Input Validation Loops, Nested Loops.
Strings: Accessing Character and Substring in Strings, Data Encryption, Strings and Number Systems.

UNIT III

List and Dictionaries: Lists, defining a list, Defining Simple Functions, Dictionaries, Defining a Dictionaries
Design with Function: Functions as Abstraction Mechanisms, Design with Recursive Functions, Higher Order Function.
Modules: Modules, Standard Modules, Packages.

UNIT IV

Object Oriented Programming: Concept of class, object and instances, Constructor, class attributes and destructors, Real time use of class in live projects, Inheritance, overlapping and overloading operators, Adding and retrieving dynamic attributes of classes, Programming using Oops support

Design with Classes: Objects and Classes, Data modeling Examples, Structuring Classes with Inheritance and Polymorphism.

UNIT V

Errors and Exceptions Text Files: Syntax Errors, Exceptions, Handling Exceptions, Raising Exceptions, User-defined Exceptions, Defining Clean-up Actions, Redefined Clean-up Actions.

File Operations: Reading config files in python, Writing log files in python, Understanding read functions, read(), readline() and readlines(), Understanding write functions, write() and writelines(), Manipulating file pointer using seek, Programming using file operations Manipulating file pointer using seek().

TEXT BOOKS:

1. Fundamentals of Python First Programs, Kenneth. A. Lambert, Cengage.
2. Python Programming: A Modern Approach, Vamsi Kurama, Pearson.

REFERENCE BOOKS:

1. Mark Lutz,"Learning Python", 5th edition, Orielly, 2013.
2. Allen Downey "Think Python, How to Think Like a Computer Scientist",2nd edition, Green Tea Press, 2015.
3. W.Chun , "Core Python Programming", 2nd Edition, Prentice Hall, 2006.
4. Kenneth A. Lambert, "Introduction to Python", 1st edition, CengageLearning, 2011.

E-Resources and other digital material:

1. Charles Severance: University of Michigan, Python for Everybody [COURSERA]. (05-01-2021), Available: https://www.coursera.org
2. Prof. Sudarshan Iyengar, IIT Ropar, Prof. Yayati Gupta, IIIT Dharwad, The Joy Of Computing Using Python [NPTEL], (05-01-2021), Available: https://nptel.ac.in/courses/106/106/106106182/#
3. Charles Russell Sevarance, University of Michigan, Python for Everybody, 2019 https://www.coursera.org/learn/python

I Year – II Semester	DIGITAL LOGIC DESIGN	L	T	P	C
		3	0	0	3

Course objectives:

- To study the basic philosophy underlying the various number systems, negative number representation, binary arithmetic, theory of Boolean algebra and map method for minimization of switching functions.
- To introduce the basic tools for design of combinational and sequential digital logic.
- To learn simple digital circuits in preparation for computer engineering.

Course outcomes:

A student who successfully fulfills the course requirements will have demonstrated:

- Classify different number systems and apply to generate various codes
- Use the concept of Boolean algebra in minimization of switching functions
- Define the Karnaugh map for a few variables and perform an algorithmic reduction of logic functions.
- Design various logic gates starting from simple ordinary gates to complex programmable logic devices.
- Design various sequential circuits starting from flipflop to registers and counters
- Develop HDL Models of Combinational Circuits.

UNIT I:

Digital Systems and Binary Numbers Digital Systems, Binary Numbers, Octal and Hexadecimal Numbers, Complements of Numbers, Signed Binary Numbers, Arithmetic addition and subtraction, 4-bit codes: BCD, EXCESS 3, alphanumeric codes, 9’s complement, 2421, etc..

UNIT II:

Concept of Boolean algebra Basic Theorems and Properties of Boolean algebra, Boolean Functions, Canonical and Standard Forms, Minterms and Maxterms. Gate level Minimization Map Method, Three-Variable K-Map, Four Variable K-Maps. Products of Sum Simplification, Sum of Products Simplification, Don’t – Care Conditions, NAND and NOR Implementation, Exclusive OR Function

UNIT III:

Combinational Logic Introduction, Analysis Procedure, Binary Adder–Subtractor, Binary Multiplier, Decoders, Encoders, Multiplexers, Demultiplexers, Priority Encoder, Code Converters, Magnitude Comparator, HDL Models of Combinational Circuits. Realization of Switching Functions Using PROM, PAL and PLA

UNIT IV:

Synchronous Sequential Logic Introduction to Sequential Circuits, Storage Elements: Latches, Flip-Flops, RS- Latch Using NAND and NOR Gates, Truth Tables. RS, JK, T and D Flip Flops, Truth and Excitation Tables, Conversion of Flip Flops.

UNIT V:

Registers and Counters Registers, Shift Registers, Ripple Counters, Synchronous Counters, Ring Counter, Johnson Counter.

Text Books:

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

Reference Books:

1) Digital Logic and Computer Design, M.Morris Mano, PEA.
2) Digital Logic Design, Leach, Malvino, Saha, TMH.
3) Modern Digital Electronics, R.P. Jain, TMH.

I Year – II Semester	DATA STRUCTURES LAB	L	T	P	C
		0	0	3	1.5

The objective of this lab is to
Demonstrate the different data structures implementation.

Course Outcomes:

- Use various searching and sorting algorithms
- Design the programs for data structures like stacks and queues by using arrays and linked list
- Design programs to demonstrate fundamental algorithmic problems like tree traversals
- Design program to implement Binary Search Tree
- Design programs to implement Graph Traversals and shortest paths

List of Experiments:

Exercise -1 (Sorting-I)
Write C program that implement Bubble sort, to sort a given list of integers in ascending order
Write C program that implement Quick sort, to sort a given list of integers in ascending order
Write C program that implement Insertion sort, to sort a given list of integers in ascending order

Exercise -2(Sorting-II)
Write C program that implement radix sort, to sort a given list of integers in ascending order
Write C program that implement merge sort, to sort a given list of integers in ascending order

Exercise -3 (Searching)
Write C program that use both recursive and non recursive functions to perform Linear search for a Key value in a given list.
Write C program that use both recursive and non recursive functions to perform Binary search for a Key value in a given list.

Exercise -4(Stack)
Write C program that implement stack (its operations) using arrays
Write C program that implement stack (its operations) using Linked list
Write a C program that uses Stack operations to evaluate postfix expression

Exercise -5(Queue)
Write C program that implement Queue (its operations) using arrays.
Write C program that implement Queue (its operations) using linked lists

Exercise -6(Singly Linked List)
Write a C program that uses functions to create a singly linked list
Write a C program that uses functions to perform insertion operation on a singly linked list
Write a C program that uses functions to perform deletion operation on a singly linked list
Write a C program to reverse elements of a single linked list.

Exercise -7(Binary Search Tree)
Write a C program to Create a BST
Write a C program to insert a node into a BST.
Write a C program to delete a node from a BST.
Write a recursive C program for traversing a binary tree in preorder, inorder and postorder.

I Year - II Semester	**APPLIED PHYSICS LAB** (Common to all branches)	L	T	P	C
		0	0	3	1.5

Course Outcomes:

- Determine the wavelength of light, radius of curvature, thickness of thin object by the process of interference of light waves.
- Describe the motion of free electrons in metals and origin of energy bands in solids.
- Determine the numerical aperture and acceptance angle of an optical fiber.
- Determine the wavelength of spectral lines, laser source, dispersive power of prism by the phenomenon of diffraction of light.
- Identify the field along axis of circular coil carrying current

List of Applied Physics Experiments:

1. Determination of thickness of thin object by wedge method.
2. Determination of radius of curvature of a given Plano convex lens by Newton’s rings.
3. Determination of wavelengths of different spectral lines in mercury spectrum using diffraction grating in normal incidence configuration.
4. Determination of dispersive power of the prism.
5. Determination of dielectric constant of a given material.
6. Determination of numerical aperture and acceptance angle of an optical fiber.
7. Determination of wavelength of Laser light using diffraction grating.
8. To determine the energy gap of a semiconductor using p-n junction diode.
9. Magnetic field along the axis of a current carrying circular coil by Stewart & Gee’s Method
10. Determination of the resistivity of semiconductor by four probe method.
11. Estimation of Planck’s constant using photoelectric effect.
12. Characteristics of Thermistor – Temperature Coefficients.
13. L- C- R Series Resonance Circuit.
14. I/V characteristics of Zener diode.
15. Verification of laws of vibrations in stretched strings – Sonometer.
16. Determination of Acceleration due to Gravity and Radius of Gyration-Compound Pendulum.
17. Rigidity modulus of material of a wire-dynamic method (Torsional pendulum).

Of the above experiments at-least 10 experiments should be completed in a semester.

References: S. Balasubramanian, M.N. Srinivasan “A Text Book of Practical Physics”- S Chand Publishers, 2017.

I Year – **II Semester**	**PYTHON PROGRAMMING LAB** (Common to all branches)	**L** **0**	**T** **0**	**P** **3**	**C** **1.5**

Course Outcomes

- Write, test, and debug simple Python programs.
- Implement Python programs with conditions and iterative loops.
- Develop Python programs step-wise by defining functions and calling them.
- Assess compound data using Python lists, tuples, and dictionaries
- Demonstrate read and write data from/to files in Python
- Solve runtime errors using exception handling

List of Experiments:

1. Write a program that asks the user for a weight in kilograms and converts it to pounds. There are 2.2 pounds in a kilogram.
2. Write a program that asks the user to enter three numbers (use three separate input statements). Create variables called total and average that hold the sum and average of the three numbers and print out the values of total and average.
3. Write a program that uses a for loop to print the numbers 8, 11, 14, 17, 20, . . ., 83, 86, 89.
4. Write a program that asks the user for their name and how many times to print it. The program should print out the user's name the specified number of times.
5. Use a for loop to print a triangle like the one below. Allow the user to specify how high the triangle should be.

   ```
   *
   **
   ***
   ****
   ```
6. Generate a random number between 1 and 10. Ask the user to guess the number and print a message based on whether they get it right or not.
7. Write a program that asks the user for two numbers and prints Close if the numbers are within .001 of each other and Not close otherwise.
8. Write a program that asks the user to enter a word and prints out whether that word contains any vowels.
9. Write a program that asks the user to enter two strings of the same length. The program should then check to see if the strings are of the same length. If they are not, the program should print an appropriate message and exit. If they are of the same length, the program should alternate the characters of the two strings. For example, if the user enters abcde and ABCDE the program should print out AaBbCcDdEe.
10. Write a program that asks the user for a large integer and inserts

commas into it according to the standard American convention for commas in large numbers. For instance, if the user enters 1000000, the output should be 1,000,000.

11. In algebraic expressions, the symbol for multiplication is often left out, as in 3x+4y or 3(x+5). Computers prefer those expressions to include the multiplication symbol, like 3*x+4*y or 3*(x+5). Write a program that asks the user for an algebraic expression and then inserts multiplication symbols where appropriate.

12. Write a program that generates a list of 20 random numbers between 1 and 100.
    a. Print the list.
    b. Print the average of the elements in the list.
    c. Print the largest and smallest values in the list.
    d. Print the second largest and second smallest entries in the list
    e. Print how many even numbers are in the list.
13. Write a program that asks the user for an integer and creates a list that consists of the factors of that integer.
14. Write a program that generates 100 random integers that are either 0 or 1. Then find the longest run of zeros, the largest number of zeros in a row. For instance, the longest run of zeros in [1,0,1,1,0,0,0,0,1,0,0] is 4.
15. Write a program that removes any repeated items from a list so that each item appears at most once. For instance, the list [1,1,2,3,4,3,0,0] would become [1,2,3,4,0].

16. Write a program that asks the user to enter a length in feet. The program should then give the user the option to convert from feet into inches, yards, miles, millimeters, centimeters, meters, or kilometers. Say if the user enters a 1, then the program converts to inches, if they enter a 2, then the program converts to yards, etc. While this can be done with if statements, it is much shorter with lists and it is also easier to add new conversions if you use lists.
17. Write a function called sum digits that is given an integer num and returns the sum of the digits of num.
18. Write a function called first diff that is given two strings and returns the first location in which the strings differ. If the strings are identical, it should return -1.
19. Write a function called number_of_factors that takes an integer and returns how many factors the number has.
20. Write a function called is sorted that is given a list and returns True if the list is sorted and False otherwise.
21. Write a function called root that is given a number x and an integer n and returns $x^{1/n}$. In the function definition, set the default value of n to 2.

22. Write a function called primes that is given a number n and returns a list of the first n primes. Let the default value of n be 100.

23. Write a function called merge that takes two already sorted lists of possibly different lengths, and merges them into a single sorted list.

    a. Do this using the sort method.

    b. Do this without using the sort method.

24. Write a program that asks the user for a word and finds all the smaller words that can be made from the letters of that word. The number of occurrences of a letter in a smaller word can't exceed the number of occurrences of the letter in the user's word.

25. Write a program that reads a file consisting of email addresses, each on its own line. Your program should print out a string consisting of those email addresses separated by semicolons.

26. Write a program that reads a list of temperatures from a file called temps.txt, converts those temperatures to Fahrenheit, and writes the results to a file called ftemps.txt.

27. Write a class called Product. The class should have fields called name, amount, and price, holding the product's name, the number of items of that product in stock, and the regular price of the product. There should be a method get price that receives the number of items to be bought and returns the cost of buying that many items, where the regular price is charged for orders of less than 10 items, a 10% discount is applied for orders of between 10 and 99 items, and a 20% discount is applied for orders of 100 or more items. There should also be a method called make purchase that receives the number of items to be bought and decreases amount by that much.

28. Write a class called Time whose only field is a time in seconds. It should have a method called convert_to_minutes that returns a string of minutes and seconds formatted as in the following example: if seconds is 230, the method should return '5:50'. It should also have a method called convert_to_hours that returns a string of hours, minutes, and seconds formatted analogously to the previous method.

29. Write a Python class to implement pow (x, n).

30. Write a Python class to reverse a string word by word.

31. Write a program to demonstrate Try/except/else.

32. Write a program to demonstrate try/finally and with/as.

II Year - I Semester	TRANSFORMS AND PARTIAL DIFFERENTIAL EQUATIONS (Common to all Branches)	L	T	P	C
		3	0	0	3

Course Objectives:

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

Course Outcomes:

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

- Apply the Laplace transform for solving differential equations(L3)
- Find or compute the Fourier series of periodic signals(L3)
- Understand the application of partial differential to estimate the maxima and minima of multiple functions (L3)
- Apply integral expressions for the forwards and inverse Fourier transform to a range of non-periodic waveforms(L3)
- Identify solution methods for partial differential equations that model physical processes(L3)

UNIT I:

Laplace Transforms: (10 hrs)

Laplace transforms of standard functions – Shifting theorems – Transforms of derivatives and integrals – Unit step function – Dirac's delta function – Inverse Laplace transforms – Convolution theorem (without proof).

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

UNIT II:

Fourier series: (10 hrs)

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: (10 hrs)

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

UNIT– IV:

Partial differentiation: (8 hrs)

Introduction–Homogeneousfunction–Euler'stheorem–Totalderivative–Chainrule–Jacobian–Functionaldependence–Taylor'sand 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 V:

PDE of first order: (10 hrs)

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.

Text Books:

1. B.S.Grewal,HigherEngineeringMathematics,43rdEdition,KhannaPublishers.
2. B.V.Ramana,HigherEngineeringMathematics,2007Edition,TataMc.GrawHillEducation.

Reference Books:

1. ErwinKreyszig,AdvancedEngineeringMathematics,10thEdition,Wiley-India.
2. Dean.G.Duffy,AdvancedEngineeringMathematicswithMATLAB,3rdEdition,CRCPress.
3. PeterO'Neil,AdvancedEngineeringMathematics,Cengage.
4. SrimanthaPal,SCBhunia,EngineeringMathematics,OxfordUniversityPress.

E-Resources:

1. https://nptel.ac.in/courses/111106139
2. https://nptel.ac.in/courses/111106111
3. https://archive.nptel.ac.in/courses/111/101/111101153/

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

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
- ❖ Determine how to represent various statements using quantifiers
- ❖ Use logical notations to formulate and reason about fundamental mathematical concepts such as sets, relations, functions and algebraic structures and Analyse the growth of functions and real world problems using various concepts like recurrence relations, graph coloring, etc
- ❖ Apply mathematical logic to solve problems, pigeonhole principle to solve real time problems
- ❖ Solve the Problems effectively mathematical ideas/results for Recurrence Relations
- ❖ Design the Model and solve real world problems using graphs and trees

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

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,

UNIT III

Algebraic Structures: Algebraic Systems, Properties, Semi Groups and Monoids, Group, Sub group and Abelian Group, Homomorphism, Isomorphism.

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

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 In homogeneous 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, Multi graphs, 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:

- ❖ Discrete Mathematical Structures with Applications to Computer Science , J.P.Tremblay and P.Manohar,TataMcGraw Hill.
- ❖ Elements of Discrete Mathematics-A Computer Oriented Approach , C.L.LiuandD.
- ❖ P.Mohapatra,3rdEdition,TataMcGrawHill.

Reference Books:

- ❖ Discrete Mathematics for Computer Scientists and Mathematician,J.L.Mott,A.KandelandT.P.Baker,2ndEdition,Prentice HallofIndia.
- ❖ Discrete Mathematical Structures,BernandKolman,RobertC.Busby and Sharon Cutler Ross,PHI.
- ❖ Discrete Mathematics and its Applications with Combinatorics and Graph Theory,K.
- ❖ H.Rosen,7thEdition,Tata Mc GrawHill.

e-Resources:

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

II Year – I Semester	**OBJECT ORIENTED PROGRAMMING THROUGH JAVA**	L	T	P	C
		3	1	0	3

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 students will be able to

- Understand the basic concepts of object oriented programming.
- Develop reusable and efficient programs using Inheritance & Polymorphism.
- Develop thread safe Java programs with appropriate Exception handling.
- Design multi-tasking applications using Multithreading
- Able to Analyze & Design the concept of Event Handling and Abstract Window Toolkit.

UNIT I

Introduction: The history and evolution of Java, Java Buzz words, object-oriented programming, Data Types, Variables and Arrays, Operators, Control Statements.

UNIT II

Classes and Objects: Concepts, methods, constructors, types of constructors, constructor overloading, usage of static, access control, this keyword, garbage collection, finalize()method, overloading, parameter passing mechanisms, final keyword, nested classes and inner classes.

UNIT III

Inheritance: Basic concepts, access specifiers, usage of super key word, method overriding, using final with Inheritance, abstract classes, dynamic method dispatch, Object class.

Interfaces: Differences between classes and interfaces, defining an interface, implementing interface, variables in interface and extending interfaces.

Packages: Creating a Package, setting CLASSPATH, Access control protection, importing packages.

Strings: Exploring the String class, String buffer class, Command-line arguments

UNIT IV
Exception Handling: Concepts of Exception handling, types of exceptions, usage of try, catch, throw, throws and finally keywords, multiple catch clauses, nested try, Built-in exceptions, creating own exception sub classes. Multithreading: The Java Thread model, thread life cycle, Thread class, Runnable interface, creating multiple threads, Synchronization, Inter Thread Communication, Deadlock.
Applets: Concepts of Applets, life cycle of an applet, creating applets

UNIT V
Event Handling: Events, Event sources, Event classes, Event Listeners, Delegation event model, handling events.
AWT: AWT Components, , File Dialog boxes, Layout Managers, Event handling model of AWT, Adapter classes, Menu, Menu bar. GUI with Swing– Swings introduction, JApplet, JFrame and JComponent, Icons and Labels, text fields, buttons – The JButton class, Check boxes, Radio buttons. Combo boxes, Tabbed Panes, Scroll Panes, Trees, and Tables

Learning Resources:
Text Book:
Java The Complete Reference - Herbert Schildt 8th Edition, Mc Graw Hill Education.

Reference Books:
1. Introduction to java programming, 7th edition by Y Daniel Liang, Pearson
2. JAVA one step ahead, Anitha Seth, B.L.Juneja, Oxford.
e-Resources:
1)https://nptel.ac.in/courses/106/105/106105191/
2)https://www.w3schools.com/java/java_data_types.asp

II Year – I Semester	COMPUTER ORGANIZATION	L	T	P	C
		3	0	0	3

Course Objectives:

The course objectives of Computer Organization are to discuss and make student familiar with

- Principles and the Implementation of Computer Arithmetic
- Operation of CPUs including RTL, ALU, Instruction Cycle and Busses
- Fundamentals of different Instruction Set Architectures and their relationship to theCPU Design
- Memory System and I/O Organization
- Principles of Operation of Multiprocessor Systems and Pipelining

Course Outcomes:

By the end of the course, the student will

- Identify the structure, function of various functional units of computer
- Describe the basic design of Computer, and its organization
- Demonstrate control unit operations and Micro Program example
- Illustrate different computer arithmetic algorithms for various arithmetic operations
- Identity and compare different methods of input-output

UNIT I

Basic Structure of Computers: Basic Organization of Computers, Historical Perspective, Bus Structures, Data Representation: Data types, Complements, Fixed Point Representation. Floating, Point Representation. Other Binary Codes, Error Detection Codes.

Computer Arithmetic: Addition and Subtraction, Multiplication Algorithms, Division Algorithms.

UNIT II

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.

Basic Computer Organization and Design: Instruction Codes, Computer Register, Computer Instructions, Instruction Cycle, Memory – Reference Instructions. Input –Output and Interrupt, Complete Computer Description.

UNIT III

Central Processing Unit: General Register Organization, STACK Organization. Instruction Formats, Addressing Modes, Data Transfer and Manipulation, Program Control, Reduced Instruction Set Computer.

Micro programmed Control: Control Memory, Address Sequencing, Micro Program example, Design of Control Unit.

UNIT IV

Memory Organization: Memory Hierarchy, Main Memory, Auxiliary Memory, Associative Memory, Cache Memory, Virtual Memory.
Input-Output Organization: Peripheral Devices, Input-Output Interface, Asynchronous data transfer, Modes of Transfer, Priority Interrupts, Direct Memory Access.

UNIT V

Multi Processors: Introduction, Characteristics of Multiprocessors, Interconnection Structures, Inter Processor Arbitration.
Pipeline: Parallel Processing, Pipelining, Instruction Pipeline, RISC Pipeline, Array Processor.

Text Books:

1) Computer System Architecture, M. Morris Mano, Third Edition, Pearson, 2008.
2) Computer Organization, Carl Hamacher, ZvonkoVranesic, SafwatZaky, 5/e, McGraw Hill, 2002.

Reference Books:

1) Computer Organization and Architecture, William Stallings, 6/e, Pearson, 2006.
2) Structured Computer Organization, Andrew S. Tanenbaum, 4/e, Pearson, 2005.
3) Fundamentals of Computer Organization and Design, Sivarama P. Dandamudi, Springer,
2006.

Web Resources:

1) https://nptel.ac.in/courses/106/105/106105163/
2) http://www.cuc.ucc.ie/CS1101/David%20Tarnoff.pdf

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

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, Failure Classification, Storage, Recovery and Atomicity, Recovery algorithm.
Indexing Techniques: B+ Trees: Search, Insert, Delete algorithms, File Organization and Indexing, Cluster Indexes, Primary and Secondary Indexes , Index data Structures, Hash Based Indexing: Tree base Indexing, Comparison of File Organizations, Indexes and Performance Tuning

Text Books:

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

Reference Books:

Introduction to Database Systems, 8/e C J Date, PEA.
Database Management System, 6/e Ramez Elmasri, Shamkant B. Navathe, PEA
Database Principles Fundamentals of Design Implementation and Management, Corlos Coronel, 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 –	**OBJECT ORIENTED PROGRAMMING**	**L**	**T**	**P**	**C**
I Semester	**WITH JAVA LAB**	**0**	**0**	**3**	**1.5**

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
- Develop GUI programs using swing controls in Java.

Exercise-1(Basics)

a) Write a JAVA program to display default value of all primitive data type of JAVA

b) 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.

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

a) Write a JAVA program to search for an element in a given list of elements using binary search mechanism.

b) Write a JAVA program to sort for an element in a given list of elements using bubble sort

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 multilevel 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

Exercise–8(Runtime Polymorphism)

a) Write a JAVA program that implements Run time polymorphism

Exercise–9(UserdefinedException)

a) Write a JAVA program for creation of Illustrating throw
b) Write a JAVA program for creation of Java Built-in Exceptions
c) 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"every3 seconds,(Repeat the same by implementing Runnable)

b) Write a program illustrating **isAlive** and **join ()**

Exercise-11(Threads continuity)

a) Write a case study on thread Synchronization after solving the above producer consumer problem

Exercise–12(Packages)

a) Write a JAVA program illustrate classpath

b) 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 JAVAprogram 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-upkey-down event user entering text in a Applet.

II Year – I Semester	COMPUTER ORGANIZATION LAB	L	T	P	C
		0	0	3	1.5

Course Objectives:

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

- Know the characteristics of various components.
- Understand the utilization of components

Course Outcomes:

At the end of this course, students will demonstrate the ability to

- Design Logic Gates and Flip flop.
- Design multiplexers and Demultiplexers
- Describe Synchronous and Asynchronous counter using flip-flops
- Solve add two 8 bit number (A+B=RESULT with a carry and without a carry).
- Solve addition, subtraction of two 16 bit numbers.

List of Experiments:

1. Realization of Boolean Expressions using Gates
2. Design and realization logic gates using universal gates
3. Design a JK Flip-Flop, Edge triggered J-K NAND Flip Flop and show its functionality Handle race condition and clock gating in your circuit.
4. Design a 4 – bit Adder / Subtractor
5. Combinational logic circuits: Implementation of Boolean functions using logic gates
6. Arithmetic operations using logic gates; Implementation of Multiplexers, Demultiplexers, Encoders, Decoders; Implementation of Boolean functions using Multiplexers/Decoders
7. Study of sequential logic circuits: Implementation of flip flops, Verify the excitation tables of various FLIP-FLOPS.
8. Design and realization a Synchronous and Asynchronous counter using flip-flops
9. Design and realization of an 8-bit parallel load and serial out shift register using flipflops
10. Implementation of counters, Design and realization a Synchronous and Asynchronous counter using flip-flops
11. Design and realization of 4x1 mux, 8x1mux using 2x1 mux

Write assembly language programs in 8086 for the following: (MASAM can also beused)

1. To add two 8 bit number (A+B=RESULT with a carry and without a carry).
2. To subtract one 8 bit number from another (A-B=RESULT with a borrow and without a borrow).
3. To find out AND, OR, NOT, XOR, NAND, NOR, XNOR of two 8 bit number.
4. To find out addition of two 16 bit numbers.
5. To find out subtraction of two 16 bit numbers.
6. To evaluate the expression a = b + c –d * e

Considering 8-bit, 16 bit and 32-bit binary numbers as b, c, d, e. Take the input in consecutive memory locations and results also Display the results by using "int xx" of 8086. Validate program for the boundary conditions.

1. To take N numbers as input. Perform the following operations on them.
    a. Arrange in ascending and descending order.
    b. Find max and minimum
    c. Find average

   Considering 8-bit, 16-bit binary numbers and 2-digit, 4 digit and 8-digit BCD numbers. Display the results by using "int xx" of 8086. Validate program for the boundary conditions.
2. To implement the above operations as procedures and call from the main procedure.
3. To find the factorial of a given number as a Procedure and call from the main program which display the result.

Note: Experiments can be done using Logic board, EasyCPU, RTSlim, Little Man Computer (LMC), Assemblers for 8085 programming, 8086 based trainer kits, MIPS simulator PCSpim, Xilinx schematic editor and simulation tools or any other choice

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

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 droping 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. 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)

6. Insert data into student table and use COMMIT, ROLLBACK and SAVEPOINT in PL/SQL block.

7. Develop a program that includes the features NESTED IF, CASE and CASE

expression. The program can be extended using the NULLIF and COALESCE functions.

8. Program development using WHILE LOOPS, numeric FOR LOOPS, nested loops using ERROR Handling, BUILT –IN Exceptions, USE defined Exceptions, RAISE- APPLICATION ERROR.

9. Program development using WHILE LOOPS, numeric FOR LOOPS, nested loops using ERROR Handling, BUILT –IN Exceptions, USE defined Exceptions, RAISE- APPLICATION ERROR.

10. Programs development using creation of procedures, passing parameters IN and OUT of PROCEDURES. Program development using creation of stored functions, invoke functions in SQL Statements and write complex functions.
11. Develop programs using features parameters in a CURSOR, FOR UPDATE CURSOR, WHERE CURRENT of clause and CURSOR Variables. 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.

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

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.arrange, np.linespace, 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.substract, 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- II Semester	**PROBABILITY AND STATISTICS** (Common to Civil,CSE & Allied Branches)	**L**	**T**	**P**	**C**
		3	**0**	**0**	**3**

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:

Up on 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)

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, $\chi^2$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 l. 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.

E-Resources:

1. https://nptel.ac.in/courses/111104032
2. https://nptel.ac.in/courses/111105090

II Year – II Semester	DESIGN AND ANALYSIS OF ALGORITHMS	L	T	P	C
		3	0	0	3

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.

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", 2ndEdition, 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 Edition, Galgotia Publications,2008.
2. S. Sridhar, "Design and Analysis of Algorithms", Oxford University Press.

II Year – II Semester	**DATA WAREHOUSING AND MINING**	**L**	**T**	**P**	**C**
		3	**0**	**0**	**3**

Course Objectives:

- To understand and implement classical models and algorithms in data warehousing and data mining.
- To analyze the data, identify the problems, and choose the relevant models and algorithms to apply.
- To assess the strengths and weaknesses of various methods and algorithms and to analyze their behavior.

Course Outcomes:

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

- Illustrate the importance of Data Warehousing, Data Mining and its functionalities and Design schema for real time data warehousing applications.
- Demonstrate on various Data Preprocessing Techniques viz. data cleaning, data integration, data transformation and data reduction and Process raw data to make it suitable for various data mining algorithms.
- Choose appropriate classification technique to perform classification, model building and evaluation
- Explain the association rule mining techniques viz. Apriori and FP Growth algorithms and analyze on frequent item sets generation.
- Identify and apply various clustering algorithm (with open source tools), interpret, evaluate and report the result.

UNIT- I

Data Warehouse and OLAP Technology: An Overview: What Is a Data Warehouse? A Multidimensional Data Model, Data Warehouse Architecture, Data Warehouse Implementation, From Data Warehousing to Data Mining. (Han &Kamber)

UNIT- II

Data Mining: Introduction, What is Data Mining?, Motivating challenges, The origins of Data Mining, Data Mining Tasks, Types of Data, Data Quality. Data Preprocessing: Aggregation, Sampling, Dimensionality Reduction, Feature Subset Selection, Feature creation, Discretization and Binarization, Variable Transformation, Measures of Similarity and Dissimilarity. (Tan &Vipin)

UNIT -III

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

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

UNIT -IV

Association Analysis: Basic Concepts and Algorithms: Problem Definition, Frequent Item Set Generation, Apriori Principle, Apriori Algorithm, Rule Generation, Compact Representation of Frequent Itemsets, FP-Growth Algorithm. (Tan &Vipin)

UNIT -V

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

Text Books:

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

Reference Books:

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

Web Resources:

1. NPTEL Online Course on Data Mining : https://onlinecourses.nptel.ac.in/noc18_cs14/preview

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

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

- Summarize language classes & grammars relationship among them with the help of Chomsky hierarchy.
- Design Automata’s and machines as Acceptors, Verifiers and Translators
- Construct finite state machines to solve problems in computing
- Illustrate deterministic and non-deterministic machines
- Assess the hierarchy of problems arising in the computer science

UNIT I

Finite Automata: Need of Automata theory, Central Concepts of Automata Theory, Automation, Finite Automation, 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 Є-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, Є-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, Types of TMs, Church's Thesis, Universal and Restricted TM, Decidable and Un-decidable Problems, Halting Problem of TMs, Post's Correspondence Problem, Modified PCP, Classes of P and NP, NP-Hard and NP-Complete Problems.

Text Books:

Introduction to Automata Theory, Languages and Computation, J. E. Hopcroft, R. Motwani and J. D. Ullman, 3rd Edition, Pearson, 2008

Theory of Computer Science-Automata, Languages and Computation, K. L. P. Mishra and N. Chandrasekharan, 3rd Edition, PHI, 2007

Reference Books:

Elements of Theory of Computation, Lewis H.P. & Papadimition C.H., Pearson /PHI

Theory of Computation, V. Kulkarni, Oxford University Press, 2013

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	**INTRODUCTION TO ARTIFICIAL INTELLIGENCE**	**L**	**T**	**P**	**C**
		3	**0**	**0**	**3**

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 provide fundamentals of machine learning

Course Outcomes:

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

- Discuss the history and foundations of Artificial Intelligence
- Apply the basic principles of AI in problem solving
- Choose the appropriate representation of Knowledge
- Choose the Perspectives and Issues in Machine Learning
- Identify issues in Decision Tree Learning

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) Tom M. Mitchell, Machine Learning, McGraw Hill Edition, 2013

Reference Books:

1) Saroj Kaushik, “Artificial Intelligence”, Cengage Learning India, 2011
2) Elaine Rich and Kevin Knight, “Artificial Intelligence”, Tata McGraw Hill
3) David Poole and Alan Mackworth, “Artificial Intelligence: Foundations for Computational Agents”, Cambridge University Press 2010.
4) Trivedi, M.C., “A Classical Approach to Artifical Intelligence”, Khanna Publishing House, Delhi.
5) Christopher Bishop, Pattern Recognition and Machine Learning (PRML) , Springer, 2007.
6) ShaiShalev-Shwartz and Shai Ben-David, Understanding Machine Learning: From Theory to Algorithms (UML) , Cambridge University Press, 2014.

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

II Year – II Semester	DATA MINING USING PYTHON LAB	L	T	P	C
		0	1	2	2

Course Objectives:

● Practical exposure on implementation of well-known data mining algorithms

● Learning performance evaluation of data mining algorithms in a supervised and an unsupervised setting.

Course Outcomes:

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

- Design a data mart or data warehouse for any organization
- Evaluate knowledge using data mining techniques and enlist various algorithms used in information analysis of Data Mining Techniques
- Demonstrate the working of algorithms for data mining tasks such as association rule mining, classification for realistic data
- Analyse on knowledge flow application on data sets and Apply the suitable visualization techniques to output analytical results
- Demonstrate the knowledge retrieve through solving problems

Note: Use python library scikit-learn wherever necessary

1. Demonstrate the following data preprocessing tasks using python libraries.
    a) Loading the dataset
    b) Identifying the dependent and independent variables
    c) Dealing with missing data

2. Demonstrate the following data preprocessing tasks using python libraries.
    a) Dealing with categorical data
    b) Scaling the features
    c) Splitting dataset into Training and Testing Sets

3. Demonstrate the following Similarity and Dissimilarity Measures using python
    a) Pearson’s Correlation
    b) Cosine Similarity
    c) Jaccard Similarity
    d) Euclidean Distance
    e) Manhattan Distance
4. Build a model using linear regression algorithm on any dataset.
5. Build a classification model using Decision Tree algorithm on iris dataset
6. Apply Naïve Bayes Classification algorithm on any dataset
7. Generate frequent itemsets using Apriori Algorithm in python and also

generate association rules for any market basket data.
8. Apply K- Means clustering algorithm on any dataset.
9. Apply Hierarchical Clustering algorithm on any dataset.
10. Apply DBSCAN clustering algorithm on any dataset.

Web Resources:

1. https://analyticsindiamag.com/data-pre-processing-in-python/
2. https://towardsdatascience.com/decision-tree-in-python-b433ae57fb93
3. https://towardsdatascience.com/calculate-similarity-the-most-relevant-metrics-in-a-nutshell-9a43564f533e
4. https://www.springboard.com/blog/data-mining-python-tutorial/
5. https://medium.com/analytics-vidhya/association-analysis-in-python-2b955d0180c

II Year – II Semester	Introduction to Artificial intelligence Lab	L	T	P	C
		0	0	3	1.5

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:

- Apply the basic principles of AI in problem solving using LISP/PROLOG
- Analyse and Formulate DFS for water jug problem, BFS for tic-tac-toe problem using LISP/PROLOG/Java
- Build TSP using heuristic approach, Simulated Annealing Algorithm using LISP/PROLOG
- Design Hill-climbing to solve 8- Puzzle Problem, Monkey Banana Problem using LISP/PROLOG
- Show an Expert System using JESS/PROLOG

List of Experiments (ARTIFICIAL INTELLIGENCE)

1. Write a program to solve 8 queens problem LISP/PROLOG
2. Solve any problem using depth first search. LISP/PROLOG
3. Solve any problem using best first search. LISP/PROLOG
4. Solve 8-puzzle problem using best first search LISP/PROLOG
5. Solve Robot (traversal) problem using means End Analysis LISP/PROLOG
6. Solve traveling salesman problem. LISP/PROLOG
7. Implementation of DFS for water jug problem using LISP/PROLOG
8. Implementation of BFS for tic-tac-toe problem using LISP/PROLOG/Java
9. Implementation of TSP using heuristic approach using Java/LISP/Prolog
10. Implementation of Simulated Annealing Algorithm using LISP/PROLOG
11. Implementation of Hill-climbing to solve 8- Puzzle Problem
12. Implementation of Monkey Banana Problem using LISP/PROLOG
13. Implementation of A* Algorithm using LISP/PROLOG

II Year – II Semester	WEB APPLICATION DEVELOPMENT LAB	L	T	P	C
		0	0	3	1.5

Course Objectives:

- To develop the skill in Creating dynamic web pages with servlets
- To provide knowledge in connecting java programs with database using JDBC.
- To develop the skill in server side programming using JSP, node.js, React.js
- To provide knowledge about MERN stack
- Testing the application on an Application Server.
- Debugging Web applications locally and remotely

Course Outcomes:

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

- Develop Single Page Applications
- Develop NodeJS & ReactJS Reusable Service
- Store the data in MySQL
- Get acquainted with the latest web application development trends in the IT industry

List of Experiments:

1. Authentication using Java Servlet
2. Authentication using JSP
3. Connect MySQL database using JSP
4. Design and development of Online Book Shop using JSP/Node.js & React.js
5. Design and development of Online Examination using JSP/Node.js & React.js
6. Design and development of online ticket reservation system using JSP/Node.js & React.js
7. Design and development of online library using JSP/Node.js & React.js
8. Design and development of online banking using JSP/Node.js & React.js
9. Design and development of online job portal using JSP/Node.js & React.js
10. Design and development of Online Auction using JSP/Node.js & React.js

Note: Students are encouraged to propose innovative ideas in the field of E-commerce as projects.

References

1. Jason Hunter, William Crawford , Java Servlet Programming, Second Edition, ,O'Reilly Media
2. Hans Bergsten, Java Server Pages, O’Reilly
3. http://www.oracle.com/technetwork/java/index-jsp-135475.html
4. http://www.oracle.com/technetwork/java/javaee/jsp/index.html

II Year - II Semester	Skill Oriented Course(APPLICATIONS OF PYTHON - PANDAS)	L	T	P	C
		1	0	2	2

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 Data Frames.
- Work with arrays, queries, and data frames
- Query Data Frame 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:

A) Pandas Data Series:

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 Data Frames:

```
                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 Data Frame 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 Data Frame.
3. Write a Pandas program to insert a new column in existing Data Frame.
4. Write a Pandas program to get list from Data Frame column headers.
5. Write a Pandas program to get list from Data Frame 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 data frame.
2. Write a Pandas program to create an index labels by using 64-bit integers, using floating-point numbers in a given data frame.

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 Data Frame column.

4.Write a Pandas program to swap the cases of a specified character column in a given Data Frame.

E)Pandas Joining and merging Data Frame:

1.Write a Pandas program to join the two given data frames along rows and assign all data.

2.Write a Pandas program to append a list of dictioneries or series to a existing Data Frame 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)Date time 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 date time.

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 data frame into groups based on school code. Also check the type of Group By object.
2. Write a Pandas program to split the following data frame by school code and get mean, min, and max value of age for each school.

H) Pandas Styling:

1. Create a data frame 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 data frame of ten rows, four columns with random values. Write a Pandas
program to highlight the maximum value in each column.

3.Create a data frame of ten rows, four columns with random values. Write a Pandas
program to highlight data frame's specific columns.

I)Excel:

1.Write a Pandas program to import excel data into a Pandas data frame.

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.