

**R23-COURSE STRUCTURE (IT)****I SEMESTER**

S.No	Course Code	Course Title	Contact Hours/Week			Credits	Scheme of Valuation		
			L	T	P		CIE	SEE	Total
<b>Theory Courses</b>									
1	23FE01	Communicative English	2	0	0	2	30	70	100
2	23FE02	Chemistry	3	0	0	3	30	70	100
3	23FE03	Linear Algebra & Calculus	3	0	0	3	30	70	100
4	23CM01	Basic Civil and Mechanical Engineering	3	0	0	3	30	70	100
5	23CS01	Introduction to Programming	3	0	0	3	30	70	100
<b>Laboratory Courses</b>									
6	23IT51	IT Workshop	0	0	2	1	30	70	100
7	23FE51	Communicative English Lab	0	0	2	1	30	70	100
8	23FE52	Chemistry Lab	0	0	2	1	30	70	100
9	23CS51	Computer Programming Lab	0	0	3	1.5	30	70	100
10	23AU01	Health and Wellness, Yoga and Sports	0	0	1	0.5	100	-	100
<b>Total</b>			<b>14</b>	<b>00</b>	<b>10</b>	<b>19</b>	370	630	1000

**II -SEMESTER**

S.No	Course Code	Course Title	Contact Hours/Week			Credits	Scheme of Valuation		
			L	T	P		CIE	SEE	Total
<b>Theory Courses</b>									
1	23FE04	Engineering Physics	3	0	0	3	30	70	100
2	23FE05	Differential Equations & Vector Calculus	3	0	0	3	30	70	100
3	23EE01	Basic Electrical and Electronics Engineering	3	0	0	3	30	70	100
4	23ME01	Engineering Graphics	2	0	2	3	30	70	100
5	23CS02	Data Structures	3	0	0	3	30	70	100
<b>Laboratory Courses</b>									
6	23FE53	Engineering Physics Lab	0	0	2	1	30	70	100
7	23EE51	Electrical & Electronics Engineering Workshop	0	0	3	1.5	30	70	100
8	23ME51	Engineering Workshop	0	0	3	1.5	30	70	100
9	23CS52	Data Structures Lab	0	0	3	1.5	30	70	100
10	23AU02	NSS/NCC/Scouts & Guides/Community Service	0	0	1	0.5	100	-	100
<b>Total</b>			<b>14</b>	<b>00</b>	<b>14</b>	<b>21</b>	370	630	1000

## III SEMESTER

S.No	Course Code	Course Title	Contact Hours/Week			Credits	Scheme of Valuation		
			L	T	P		CIE	SEE	Total
<b>Theory Courses</b>									
1	23FE11	Discrete Mathematics & Graph Theory	3	0	0	3	30	70	100
2	23HS01	UHV 2 - Understanding Harmony and Ethical Human Conduct	2	1	0	3	30	70	100
3	23IT01	Digital Logic & Computer Organization	3	0	0	3	30	70	100
4	23CS04	Advanced Data Structures & Algorithm Analysis	3	0	0	3	30	70	100
5	23CS05	Object Oriented Programming through Java	3	0	0	3	30	70	100
<b>Laboratory Courses</b>									
6	23CS53	Advanced Data Structures & Algorithm Analysis Lab	0	0	3	1.5	30	70	100
7	23CS54	Object Oriented Programming through Java Lab	0	0	3	1.5	30	70	100
8	23CSS2	Full Stack Development-I	0	1	2	2	30	70	100
<b>Total</b>			<b>14</b>	<b>02</b>	<b>08</b>	<b>20</b>	240	560	800

## IV SEMESTER

S.No	Course Code	Course Title	Contact Hours/Week			Credits	Scheme of Valuation		
			L	T	P		CIE	SEE	Total
<b>Theory Courses</b>									
1	23ME09	Optimization Techniques	2	0	0	2	30	70	100
2	23FE10	Probability & Statistics	3	0	0	3	30	70	100
3	23CS06	Operating Systems	3	0	0	3	30	70	100
4	23CS03	Database Management Systems	3	0	0	3	30	70	100
5	23IT02	Software Engineering	3	0	0	3	30	70	100
<b>Laboratory Courses</b>									
6	23IT52	Operating Systems & Software Engineering Lab	0	0	3	1.5	30	70	100
7	23CS56	Database Management Systems Lab	0	0	3	1.5	30	70	100
8	23CSS1	Python Programming	0	1	2	2	30	70	100
9	23ME57	Design Thinking & Innovation	1	0	2	2	30	70	100
10	23MC01	Environmental Science	2	0	0	-	30	-	030
<b>Total</b>			<b>17</b>	<b>01</b>	<b>10</b>	<b>21</b>	300	630	930

Mandatory Community Service Project Internship of 08 weeks' duration during summer vacation.

## V SEMESTER

S. No	Course Code	Course Title	Contact Hours/Week			Credits	Scheme of Valuation		
			L	T	P		CIE	SEE	Total
<b>Theory Courses</b>									
1	23IT04	Advanced Java	3	0	0	3	30	70	100
2	23CS07	Computer Networks	3	0	0	3	30	70	100
3	23CS12	Automata Theory & Compiler Design	3	0	0	3	30	70	100
4	23IT03	1. Object Oriented Analysis and Design	3	0	0	3	30	70	100
	23IT05	2. Cyber Security							
	23AD02	3. Artificial Intelligence							
	23AD04	4. Data Warehousing & Data Mining 12-weekMOOC Swayam/ NPTEL course recommended by the BoS							
5		Open Elective- I	3	0	0	3	30	70	100
<b>Laboratory Courses</b>									
6	23IT54	Advanced Java Lab	0	0	3	1.5	30	70	100
7	23CS58	Computer Networks Lab	0	0	3	1.5	30	70	100
8	23ITS1	Python with Django	0	1	2	2	30	70	100
9	23IT53	User Interface Design using Flutter / SWAYAM Plus - Android Application Development (with Flutter)	0	0	2	1	30	70	100
10	23PI01	Evaluation of Community Service Internship	-	-	-	2	-	50	50
<b>Total</b>			<b>15</b>	<b>01</b>	<b>10</b>	<b>23</b>	<b>270</b>	<b>680</b>	<b>950</b>

## VI SEMESTER

S. No	Course Code	Course Title	Contact Hours/Week			Credits	Scheme of Valuation		
			L	T	P		CIE	SEE	Total
<b>Theory Courses</b>									
1	23CS11	Cloud Computing	3	0	0	3	30	70	100
2	23CS13	Cryptography & Network Security	3	0	0	3	30	70	100
3	23AM01	Machine Learning	3	0	0	3	30	70	100
4	23IT06	1. Software Testing Methodologies	3	0	0	3	30	70	100
	23IT09	2. Augmented Reality & Virtual Reality							
	23IT08	3. DevOps							
	23IT10	4. Generative AI							
12-week MOOC Swayam/NPTEL course recommended by the BoS									
5	23IT07	1. Software Project Management	3	0	0	3	30	70	100
	23CS14	2. Mobile Adhoc Networks							
	23AM05	3. Natural Language Processing							
	23CS09	4. Distributed Operating System							
12-week MOOC Swayam/NPTEL Course recommended by the BoS									
6		Open Elective- II	3	0	0	3	30	70	100
<b>Laboratory Courses</b>									
7	23CS59	Cloud Computing Lab	0	0	3	1.5	30	70	100
8	23AM51	Machine Learning Lab	0	0	3	1.5	30	70	100
9	23HSS1	Soft skills // SWAYAM Plus - 21st Century Employability Skills	0	1	2	2	30	70	100
10	23MC04	Technical Paper Writing & IPR	2	0	0	-	30	-	030
<b>Total</b>			<b>20</b>	<b>01</b>	<b>08</b>	<b>23</b>	<b>300</b>	<b>630</b>	<b>930</b>
<b>Mandatory Industry Internship / Mini Project of 08 weeks duration during summer vacation.</b>									

**LAKIREDDY BALI REDDY COLLEGE OF ENGINEERING (AUTONOMOUS)**  
**OPEN ELECTIVES**

<b>Course Code</b>	<b>Course Name</b>	<b>Offered to the branches</b>
23AD81	Introduction to Artificial Intelligence	ASE, CE, ECE,EEE & ME
23AD82	Fundamentals of Data Science	ASE, CE, ECE,EEE & ME
23AD83	Introduction to Cloud Computing	ASE, CE, ECE,EEE & ME
23AD84	Data Analytics	ASE, CE, ECE,EEE & ME
23AE81	PRINCIPLES OF FLIGHT	AI&DS, CE,CSE, CSE(AI&ML), ECE,EEE,IT & ME
23AE82	SPACE SCIENCE	AI&DS, CE,CSE, CSE(AI&ML), ECE,EEE,IT & ME
23AE83	AIRCRAFT INSTRUMENTATION	AI&DS, CE,CSE, CSE(AI&ML), ECE,EEE,IT & ME
23AE84	AIR TRANSPORTATION SYSTEMS	AI&DS, CE,CSE, CSE(AI&ML), ECE,EEE,IT & ME
23AM81	Python Programming for AI & ML	ASE, CE, ECE,EEE & ME
23AM82	AI in healthcare	ASE, CE, ECE,EEE & ME
23AM83	Fundamentals of Machine Learning	ASE, CE, ECE,EEE & ME
23AM84	Introduction to Deep learning	ASE, CE, ECE,EEE & ME
23CE81	Disaster Management	ASE,AI&DS,CSE, CSE(AI&ML), ECE,EEE,IT & ME
23CE82	Climate change impact on Eco system	ASE,AI&DS,CSE, CSE(AI&ML), ECE,EEE,IT & ME
23CE83	Environmental Sanitation	ASE,AI&DS,CSE, CSE(AI&ML), ECE,EEE,IT & ME
23CE84	Introduction to Remote Sensing and GIS	ASE,AI&DS,CSE, CSE(AI&ML), ECE,EEE,IT & ME
23CE85	Water Supply Systems	ASE,AI&DS,CSE, CSE(AI&ML), ECE,EEE,IT & ME
23CE86	Sustainability in Engineering Practices	ASE,AI&DS,CSE, CSE(AI&ML), ECE,EEE,IT & ME
23CS81	Introduction to Java Programming	ASE, CE, ECE,EEE & ME
23CS82	Principles of Operating Systems	ASE, CE, ECE,EEE & ME
23CS83	Principles of Database Management Systems	ASE, CE, ECE,EEE & ME
23CS83	Principles of Database Management Systems	ASE, CE, ECE,EEE & ME
23CS84	IoT based smart Systems	ASE, CE, ECE,EEE & ME
23EC81	Linear and Digital IC Applications	ASE,AI&DS, CE,CSE, CSE(AI&ML), EEE,IT & ME
23EC82	Principles of communications	ASE,AI&DS, CE,CSE, CSE(AI&ML), EEE,IT & ME
23EC83	Fundamentals of VLSI Design	ASE,AI&DS, CE,CSE, CSE(AI&ML), EEE,IT & ME
23EC84	Principles of Cellular & Mobile communications	ASE,AI&DS, CE,CSE, CSE(AI&ML), EEE,IT & ME
23EC85	Fundamentals of Satellite Communications	ASE,AI&DS, CE,CSE, CSE(AI&ML), EEE,IT & ME

## LAKIREDDY BALI REDDY COLLEGE OF ENGINEERING (AUTONOMOUS)

Course Code	Course Name	Offered to the branches
23EE81	Basic Control System	ASE,AI&DS, CE,CSE, CSE(AI&ML), ECE,IT & ME
23EE82	Basic Electrical Measurements	ASE,AI&DS, CE,CSE, CSE(AI&ML), ECE,IT & ME
23EE83	Utilization of Electrical Energy	ASE,AI&DS, CE,CSE, CSE(AI&ML), ECE,IT & ME
23EE84	Electric Vehicles	ASE,AI&DS, CE,CSE, CSE(AI&ML), ECE,IT & ME
23EE85	Concepts of Energy Auditing & Management	ASE,AI&DS, CE,CSE, CSE(AI&ML), ECE,IT & ME
23EE86	Electrical Wiring Estimation and Costing	ASE,AI&DS, CE,CSE, CSE(AI&ML), ECE,IT & ME
23IT81	Computer System Architecture	ASE, CE, ECE,EEE & ME
23IT82	Introduction to Programming in Java	ASE, CE, ECE,EEE & ME
23IT83	Principles of Software Engineering	ASE, CE, ECE,EEE & ME
23ME81	Sustainable Energy Technologies	ASE,AI&DS, CE,CSE, CSE(AI&ML), ECE,EEE & IT
23ME82	Introduction to Industrial Robotics	ASE,AI&DS, CE,CSE, CSE(AI&ML), ECE,EEE & IT
23ME83	Applied Operations Research	ASE,AI&DS, CE,CSE, CSE(AI&ML), ECE,EEE & IT
23ME84	Entrepreneurship	ASE,AI&DS, CE,CSE, CSE(AI&ML), ECE,EEE & IT
23ME85	Additive Manufacturing	ASE,AI&DS, CE,CSE, CSE(AI&ML), ECE,EEE & IT
23ME86	Vehicle Technology	ASE,AI&DS, CE,CSE, CSE(AI&ML), ECE,EEE & IT

B.Tech. (I Sem.)

23FE01-COMMUNICATIVE ENGLISH

L	T	P	Cr.
2	0	0	2

**Pre-requisites:** Nil

**Course Objectives:** The main objective of introducing this course, Communicative English, is to facilitate effective listening, Reading, Speaking and Writing skills among the students. It enhances the same in their comprehending abilities, oral presentations, reporting useful information and providing knowledge of grammatical structures and vocabulary. This course helps the students to make them effective in speaking and writing skills and to make them industry ready

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

- CO1:** Understand the context, topic, and pieces of specific information from social or Transactional dialogues. (**Understand-L2**)
- CO2:** Apply grammatical structures to formulate sentences and correct word forms. (**Apply-L3**)
- CO3:** Use discourse markers to speak clearly on a specific topic in informal discussions. (**Apply-L3**)
- CO4:** Read / Listen the texts and write summaries based on global comprehension of these texts. (**Understand-L2**)
- CO5:** Prepare a coherent paragraph, essay, and resume. (**Apply-L3**)

**Unit – I****Lesson: HUMAN VALUES: Gift of Magi (Short Story)**

**Listening:** Identifying the topic, the context and specific pieces of information by listening to short audio texts and answering a series of questions.

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

**Reading:** Skimming to get the main idea of a text; scanning to look for specific pieces of Information.

**Writing:** Mechanics of Writing-Capitalization, Spellings, Punctuation-Parts of Sentences.

**Grammar:** Parts of Speech, Basic Sentence Structures-forming questions

**Vocabulary:** Synonyms, Antonyms, Affixes (Prefixes/Suffixes), Root words.

**Unit – II****Lesson: NATURE: The Brook by Alfred Tennyson (Poem)**

**Listening:** Answering a series of questions about main ideas and supporting ideas after Listening to audio texts.

**Speaking:** Discussion in pairs/small groups on specific topics followed by short structure talks.

**Reading:** Identifying sequence of ideas; recognizing verbal techniques that help to link the ideas in a paragraph together.

**Writing:** Structure of a paragraph - Paragraph writing (specific topics)

**Grammar:** Cohesive devices - linkers, use of articles and zero article; prepositions.

**Vocabulary:** Homonyms, Homophones, Homographs.

**Unit – III****Lesson: BIOGRAPHY: Elon Musk**

**Listening:** Listening for global comprehension and summarizing what is listened to.

**Speaking:** Discussing specific topics in pairs or small groups and reporting what is discussed.

- Reading:** Reading a text in detail by making basic inferences -recognizing and interpreting specific context clues; strategies to use text clues for comprehension.
- Writing:** Summarizing, Note-making, paraphrasing
- Grammar:** Verbs - tenses; subject-verb agreement; Compound words, Collocations
- Vocabulary:** Compound words, Collocations

#### Unit – IV

##### Lesson: **INSPIRATION: The Toys of Peace by Saki**

- Listening:** Making predictions while listening to conversations/ transactional dialogues without video; listening with video.
- Speaking:** Role plays for practice of conversational English in academic contexts (formal and informal) - asking for and giving information/directions.
- Reading:** Studying the use of graphic elements in texts to convey information, reveal trends/ patterns /relationships, communicate processes or display complicated data.
- Writing:** Letter Writing: Official Letters, Resumes
- Grammar:** Reporting verbs, Direct & Indirect speech, Active & Passive Voice
- Vocabulary:** Words often confused, Jargons.

#### Unit – V

##### Lesson: **MOTIVATION: The Power of Intrapersonal Communication (An Essay)**

- Listening:** Identifying key terms, understanding concepts and answering a series of relevant questions that test comprehension.
- Speaking:** Formal oral presentations on topics from academic contexts
- Reading:** Reading comprehension.
- Writing:** Writing structured essays on specific topics.
- Grammar:** Editing short texts –identifying and correcting common errors in grammar and usage (articles, prepositions, tenses, subject verb agreement)
- Vocabulary:** Technical Jargons

#### Textbooks:

1. Pathfinder: Communicative English for Undergraduate Students, 1st Edition, Orient Black Swan, 2023 (Units 1,2 & 3)
2. Empowering with Language by Cengage Publications, 2023 (Units 4 & 5)

#### Reference Books:

1. Dubey, Sham Ji & Co. English for Engineers, Vikas Publishers, 2020
2. Bailey, Stephen. Academic writing: A Handbook for International Students. Routledge.
3. Murphy, Raymond. English Grammar in Use, Fourth Edition, Cambridge University Press, 2019.

#### Web Resources:

##### GRAMMAR:

1. [www.bbc.co.uk/learningenglish](http://www.bbc.co.uk/learningenglish)
2. <https://dictionary.cambridge.org/grammar/british-grammar/>
3. [www.eslpod.com/index.html](http://www.eslpod.com/index.html)
4. <https://www.learngrammar.net/>
5. <https://english4today.com/english-grammar-online-with-quizzes/>
6. <https://www.talkenglish.com/grammar/grammar.aspx>

##### VOCABULARY

1. <https://www.youtube.com/c/DailyVideoVocabulary/videos>
2. [https://www.youtube.com/channel/UC4cmBAit8i\\_NJZE8qK8sfpA](https://www.youtube.com/channel/UC4cmBAit8i_NJZE8qK8sfpA)

L	T	P	Cr.
3	0	0	3

B.Tech. (I Sem.)

23FE02-CHEMISTRY

**Pre-requisites:** Nil**Course Objectives:**

- To enable the students to understand the fundamental concepts of chemistry and to provide them with the knowledge of industrial problems and finding the solutions.
- To strengthen the basic concepts of bonding models, advanced engineering materials, electrochemistry, batteries and polymers.
- To introduce instrumental methods and their applications.

**Course Outcomes:** At the end of the course, students will be able to,**CO1:** Understand the fundamentals of quantum mechanics and molecular orbital energy diagrams for molecules. **(Understand-L2)****CO2:** Summarize the suitability of advanced materials like semiconductors, superconductors, super capacitors and Nano materials, in advanced fields. **(Understand-L2)****CO3:** Apply Nernst equation in calculating cell potentials and understand conduct metric, potentiometric titrations, electrochemical sensors and compare batteries for different applications. **(Understand-L2)****CO4:** Outline the importance of polymers and conducting polymers in advanced technologies. **(Understand-L2)****CO5:** Understand the fundamentals of UV-Visible, IR spectroscopic techniques and basic principles of chromatographic techniques. **(Understand-L2)****UNIT – I:****Structure and Bonding Models:**

Fundamentals of Quantum mechanics, Schrodinger Wave equation, significance of  $\Psi$  and  $\Psi^2$ , particle in one dimensional box, molecular orbital theory – bonding in homo- and hetero nuclear diatomic molecules – energy level diagrams of O<sub>2</sub> and CO, etc.  $\pi$ -molecular orbitals of butadiene and benzene, calculation of bond order.

**UNIT – II:****Modern Engineering materials:****Semiconductors:** Introduction, basic concept, applications.**Super conductors:** Introduction basic concept, applications.**Super capacitors:** Introduction, Basic Concept, Classification and applications.**Nano materials:** Introduction, classification, properties and applications of Fullerenes, carbon nano tubes and Graphines nano particles.**UNIT – III:****Electrochemistry and Applications**

Electrochemical cell, Nernst equation, cell potential calculations and numerical problems, potentiometry- potentiometric titrations (redox titrations), concept of conductivity, conductivity cell, conductometric titrations (acid-base titrations).

**Electrochemical sensors** – potentiometric sensors with examples, amperometric sensors with examples.**Primary cells** – Zinc-air battery, Secondary cells –lithium-ion batteries- working of the batteries including cell reactions; Fuel cells, hydrogen-oxygen fuel cell– working of the cells. Polymer Electrolyte Membrane Fuel cells (PEMFC).

**UNIT – IV:**

**Polymer Chemistry**

Introduction to polymers, functionality of monomers, chain growth and step growth polymerization, coordination polymerization, with specific examples and mechanisms of polymer formation.

**Plastics** –Thermo and Thermosetting plastics, Preparation, properties and applications of – PVC, Teflon, Bakelite, Nylon-6,6, carbon fibres.

**Elastomers**– Buna-S, Buna-N–preparation, properties and applications.

**Conducting polymers** – polyacetylene, polyaniline, – mechanism of conduction and applications. Bio-Degradable polymers - Poly Glycolic Acid (PGA), Polyl Lactic Acid (PLA).

**UNIT – V:**

**Instrumental Methods and Applications**

Electromagnetic spectrum. Absorption of radiation: Beer-Lambert's law. UV-Visible Spectroscopy, electronic transition, Instrumentation, IR spectroscopies, fundamental modes and selection rules, Instrumentation. Chromatography-Basic Principle, Classification-HPLC: Principle, Instrumentation and Applications.

**Textbooks:**

1. Jain and Jain, Engineering Chemistry, 16/e, DhanpatRai, 2013.
2. Peter Atkins, Julio de Paula and James Keeler, Atkins' Physical Chemistry, 10/e, Oxford University Press, 2010.

**Reference Books:**

1. Skoog and West, Principles of Instrumental Analysis, 6/e, Thomson, 2007.
2. J.D. Lee, Concise Inorganic Chemistry, 5th Edition, Wiley Publications, Feb.2008
3. Textbook of Polymer Science, Fred W. Billmeyer Jr, 3rd Edition

L	T	P	Cr.
3	0	0	3

B.Tech. (I Sem.)

23FE03-LINEAR ALGEBRA &amp; CALCULUS

**Pre-requisites:** Basics of Calculus

**Course Educational Objective:** 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 this course, the student will be able to

- CO1:** Apply matrix algebra techniques to solve engineering problems. (**Apply-L3**)
- CO2:** Use Eigen values and Eigen vectors concept to find nature of quadratic form, inverse and powers of matrix. (**Apply-L3**)
- CO3:** Expand various functions using Mean value theorems. (**Understand-L2**)
- CO4:** Understand the concepts of functions of several variables which are useful in optimization. (**Understand-L2**)
- CO5:** Evaluate areas and volumes by using double and triple integrals. (**Apply-L3**)

#### UNIT –I:

##### Matrices

Rank of a matrix by echelon form, normal form. Cauchy–Binet formulae (without proof). Inverse of Non- singular matrices by Gauss-Jordan method, System of linear equations: Solving system of Homogeneous and Non-Homogeneous equations by Gauss elimination method, Jacobi and Gauss Seidel Iteration Methods.

#### UNIT –II

##### Eigen values, Eigenvectors and Orthogonal Transformation

Eigen values, Eigenvectors and their properties, Diagonalization of a matrix, Cayley-Hamilton Theorem (without proof), finding inverse and power of a matrix by Cayley-Hamilton Theorem, Quadratic forms and Nature of the Quadratic Forms, Reduction of Quadratic form to canonical forms by Orthogonal Transformation.

#### UNIT – III:

##### Calculus

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

#### UNIT –IV

##### Partial differentiation and Applications (Multi variable calculus)

Functions of several variables: Continuity and Differentiability, Partial derivatives, total derivatives, chain rule, Directional derivative, Taylor’s and Maclaurin’s series expansion of functions of two variables. Jacobians, Functional dependence, maxima and minima of functions of two variables, method of Lagrange multipliers.

**UNIT – V**

**Multiple Integrals (Multi variable Calculus)**

Double integrals, triple integrals, change of order of integration, change of variables to polar, cylindrical and spherical coordinates. Finding areas (by double integrals) and volumes (by double integrals and triple integrals).

**Textbooks:**

1. Higher Engineering Mathematics, B. S. Grewal, Khanna Publishers, 2017, 44th Edition
2. Advanced Engineering Mathematics, Erwin Kreyszig, John Wiley & Sons, 2018, 10th edition.

**Reference Books:**

1. Thomas Calculus, George B. Thomas, Maurice D. Weir and Joel Hass, Pearson Publishers, 2018, 14th Edition.
2. Advanced Engineering Mathematics, R. K. Jain and S. R. K. Iyengar, Alpha Science International Ltd., 2021 5th Edition (9th reprint).
3. Advanced Modern Engineering Mathematics, Glyn James, Pearson publishers, 2018, 5th edition.
4. Advanced Engineering Mathematics, Micheael Greenberg, , Pearson publishers, 9<sup>th</sup> edition
5. Higher Engineering Mathematics, H. K Das, Er. Rajnish Verma, S. Chand Publications, 2014, Third Edition (Reprint 2021)

B.Tech. (I Sem.)

23CM01- BASIC CIVIL AND MECHANICAL  
ENGINEERING

L	T	P	Cr.
3	0	0	3

**Prerequisite:** Nil**PART-A: BASIC CIVIL ENGINEERING****Course Objectives:**

- Get familiarized with the scope and importance of Civil Engineering sub-divisions.
- Introduce the preliminary concepts of surveying.
- Acquire preliminary knowledge on Transportation and its importance in nation's economy.
- Get familiarized with the importance of quality, conveyance and storage of water.
- Introduction to basic civil engineering materials and construction techniques.

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

- CO1:** Describe various sub-divisions of Civil Engineering and to appreciate their role in societal development. **(Understand-L2)**
- CO2:** Outline the concepts of surveying and obtain the theoretical measurement of distances, angles and levels through surveying. **(Understand-L2)**
- CO3:** Classify the various materials used in construction and highway engineering and identify their appropriate usage as per the needs. **(Understand-L2)**
- CO4:** Illustrate the fundamental principles involved in transportation network system, their individual components and their engineering importance. **(Understand-L2)**
- CO5:** Explain the quality parameters of various water sources and functions of selected water storage and conveyance structures. **(Understand-L2)**

**UNIT I****Basics of Civil Engineering:**

Role of Civil Engineers in Society- Various Disciplines of Civil Engineering- Structural Engineering- Geo-technical Engineering- Transportation Engineering - Hydraulics and Water Resources Engineering - Environmental Engineering-Scope of each discipline - Building Construction and Planning- Construction Materials-Cement - Aggregate -Bricks- Cement concrete- Steel. Introduction to Prefabricated construction Techniques.

**UNIT II**

**Surveying:** Objectives of Surveying- Horizontal Measurements- Angular Measurements Introduction to Bearings, Leveling instruments used for leveling -Simple problems on leveling and bearings-Contour mapping.

**UNIT III****Transportation Engineering**

Importance of Transportation in Nation's economic development- Types of Highway Pavements- Flexible Pavements and Rigid Pavements -Simple Differences. Basics of Harbour, Tunnel, Airport, and Railway Engineering.

**Water Resources and Environmental Engineering:** Introduction, Sources of water- Quality of water- Specifications- Introduction to Hydrology-Rainwater Harvesting-Water Storage and Conveyance Structures (Simple introduction to Dams and Reservoirs).

**Textbooks:**

1. Basic Civil Engineering, M.S.Palanisamy, , Tata Mcgraw Hill publications (India) Pvt. Ltd. Fourth Edition.
2. Introduction to Civil Engineering, S.S. Bhavikatti, New Age International Publishers. 2022. First Edition.

3. Basic Civil Engineering, Satheesh Gopi, Pearson Publications, 2009, First Edition.

**Reference Books:**

1. Surveying, Vol- I and Vol-II, S.K. Duggal, Tata McGraw Hill Publishers 2019. Fifth Edition.
2. Hydrology and Water Resources Engineering, Santosh Kumar Garg, Khanna Publishers, Delhi. 2016
3. Irrigation Engineering and Hydraulic Structures - Santosh Kumar Garg, Khanna Publishers, Delhi 2023. 38th Edition.
4. Highway Engineering, S. K. Khanna, C.E.G. Justo and Veeraraghavan, Nemchand and Brothers Publications 2019. 10th Edition.
5. Indian Standard DRINKING WATER — SPECIFICATION IS 10500-2012.

**PART-B: BASIC MECHANICAL ENGINEERING**

**Course Objectives:** The students after completing the course are expected to

- Get familiarized with the scope and importance of Mechanical Engineering in different sectors and industries.
- Explain different engineering materials and different manufacturing processes.
- Provide an overview of different thermal and mechanical transmission systems and introduce basics of robotics and its applications.

**Course Outcomes:** On completion of the course, the student should be able to

**CO1:** Summarize the different manufacturing processes. (**Remember-L1**)

**CO2:** Explain the basics of thermal engineering and its applications. (**Understand-L2**)

**CO3:** Illustrate the working of different mechanical power transmission systems and power plants (**Understand-L2**).

**CO4:** Describe the basics of robotics and its applications (**Understand-L2**)

**UNIT I**

**Introduction to Mechanical Engineering:** Role of Mechanical Engineering in Industries and Society- Technologies in different sectors such as Energy, Manufacturing, Automotive, Aerospace, and Marine sectors.

**Engineering Materials** - Metals-Ferrous and Non-ferrous, Ceramics, Composites, Smart materials.

**UNIT II**

**Manufacturing Processes:** Principles of Casting, Forming, joining processes, Machining, Introduction to CNC machines, 3D printing, and Smart manufacturing.

**Thermal Engineering** – Working principle of Boilers, Otto cycle, Diesel cycle, Refrigeration and air-conditioning cycles, IC engines, 2-Stroke and 4-Stroke engines, SI/CI Engines, Components of Electric and Hybrid Vehicles.

**UNIT III**

**Power plants** – Working principle of Steam, Diesel, Hydro, Nuclear power plants. Mechanical Power Transmission - Belt Drives, Chain, Rope drives, Gear Drives and their applications.

**Introduction to Robotics** - Joints & links, configurations, and applications of robotics.

**(Note: The subject covers only the basic principles of Civil and Mechanical Engineering systems. The evaluation shall be intended to test only the fundamentals of the subject.)**

**Textbooks:**

1. Internal Combustion Engines by V.Ganesan, By Tata McGraw Hill publications (India) Pvt. Ltd.
2. A text book of Theory of Machines by S.S. Rattan, Tata McGraw Hill Publications, (India) Pvt. Ltd.
3. An introduction to Mechanical Engg by Jonathan Wicker and Kemper Lewis, Cengage learning India Pvt. Ltd.

**Reference Books:**

1. G. Shanmugam and M.S.Palanisamy, Basic Civil and the Mechanical Engineering, Tata McGraw Hill publications (India) Pvt. Ltd.
2. Thermal Engineering by Mahesh M Rathore Tata McGraw Hill publications (India) Pvt. Ltd.
3. 3D printing & Additive Manufacturing Technology- L. Jyothish Kumar, Pulak M Pandey, Springer publications
4. Appuu Kuttan KK, Robotics, I.K. International Publishing House Pvt. Ltd. Volume-I

B.Tech. (I Sem.)

23CS01-INTRODUCTION TO PROGRAMMING

L	T	P	Cr.
3	0	0	3

**Pre-requisite** : Mathematics, Basic Computer concepts

**Course Objectives:**

- To introduce students to the fundamentals of computer programming.
- To provide hands-on experience with coding and debugging.
- To foster logical thinking and problem-solving skills using programming.
- To familiarize students with programming concepts such as data types, control structures, functions, and arrays.
- To encourage collaborative learning and teamwork in coding projects.

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

**CO1:** Understand basics of computers, concept of algorithms and flowcharts. (**Understand-L2**)

**CO2:** Understand the features of C language. (**Understand-L2**)

**CO3:** Interpret the problem and develop an algorithm to solve it. (**Apply-L3**)

**CO4:** Implement various algorithms using the C programming language. (**Apply-L3**)

**CO5:** Develop skills required for problem-solving and optimizing the code (**Apply-L3**)

**UNIT – I**

**Introduction to Programming and Problem Solving**

History of Computers, Basic organization of a computer: ALU, input-output units, memory, program counter, Introduction to Programming Languages, Basics of a Computer Program, Algorithms, flowcharts (Using Dia Tool), pseudo code. Introduction to Compilation and Execution, Primitive Data Types, Variables, and Constants, Basic Input and Output, Operations, Type Conversion, and Casting.

**Problem solving techniques:** Algorithmic approach, characteristics of algorithm, Problem solving strategies: Top-down approach, Bottom-up approach, Time and space complexities of algorithms.

**UNIT – II**

**Control Structures**

Simple sequential programs Conditional Statements (if, if-else, switch), Loops (for, while, do while) Break and Continue.

**UNIT – III**

**Arrays and Strings**

Arrays indexing, memory model, programs with array of integers, two dimensional arrays, Introduction to Strings.

**UNIT – IV**

**Pointers & User Defined Data types**

Pointers, dereferencing and address operators, pointer and address arithmetic, array manipulation using pointers, User-defined data types-Structures and Unions.

**UNIT – V**

**Functions & File Handling**

Introduction to Functions, Function Declaration and Definition, Function call Return Types and Arguments, modifying parameters inside functions using pointers, arrays as parameters. Scope and Lifetime of Variables, Basics of File Handling

**Textbooks:**

1. "The C Programming Language", Brian W. Kernighan and Dennis M. Ritchie, PrenticeHall, 1988
2. Schaum's Outline of Programming with C, Byron S Gottfried, McGraw-Hill Education, 1996

**Reference Books:**

1. Computing fundamentals and C Programming, Balagurusamy, E., McGraw-Hill Education, 2008.
2. Programming in C, Rema Theraja, Oxford, 2016, 2nd edition
3. C Programming, A Problem Solving Approach, Forouzan, Gilberg, Prasad, CENGAGE, 3rd edition

L	T	P	Cr.
0	0	2	1

B.Tech. (I Sem.)

23IT51-IT WORKSHOP

Pre-requisite : NIL

**Course Objectives:**

- To introduce the internal parts of a computer, peripherals, I/O ports, connecting cables
- To demonstrate configuring the system as Dual boot both Windows and other Operating Systems Viz. Linux, BOSS
- To teach basic command line interface commands on Linux.
- To teach the usage of Internet for productivity and self-paced life-long learning
- To introduce Compression, Multimedia and Antivirus tools and Office Tools such as Word processors, Spread sheets and Presentation tools.

**Course Outcomes (COs):** At the end of this course, the student will be able to:

**CO1:** Identify the components of a PC and troubleshooting the malfunctioning of PC.  
(Apply-L3)

**CO2:** Develop presentation /documentation using Office tools and LaTeX. (Apply-L3)

**CO3:** Build dialogs and documents using ChatGPT. (Apply-L3)

**CO4:** Improve individual / teamwork skills, communication and report writing skills with ethical values.

**PC Hardware & Software Installation**

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

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

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

**Task 4:** Every student should install Linux on the computer. This computer should have windows installed. The system should be configured as dual boot (VMWare) with both Windows and Linux. Lab instructors should verify the installation and follow it up with a Viva

**Task 5:** Every student should install BOSS on the computer. The system should be configured as dual boot (VMWare) with both Windows and BOSS. Lab instructors should verify the installation and follow it up with a Viva

**Internet & World Wide Web**

**Task1:** Orientation & Connectivity Boot Camp: Students should get connected to their Local Area Network and access the Internet. In the process they configure the TCP/IP setting. Finally students should demonstrate, to the instructor, how to access the websites and email. If there is no internet connectivity preparations need to be made by the instructors to simulate the WWW on the LAN.

**Task 2:** Web Browsers, Surfing the Web: Students customize their web browsers with the LAN proxy settings, bookmarks, search toolbars and pop up blockers. Also, plug-ins like Macromedia Flash and JRE for applets should be configured.

**Task 3:** Search Engines & Netiquette: Students should know what search engines are and how to use the search engines. A few topics would be given to the students for which they need to search on Google. This should be demonstrated to the instructors by the student.

**Task 4:** Cyber Hygiene: Students would be exposed to the various threats on the internet and would be asked to configure their computer to be safe on the internet. They need to customize their browsers to block pop ups, block active x downloads to avoid viruses and/or worms.

### **LaTeX and WORD**

**Task 1:** Word Orientation: The mentor needs to give an overview of La TeX and Microsoft (MS) office or equivalent (FOSS) tool word: Importance of La TeX and MS office or equivalent (FOSS) tool Word as word Processors, Details of the four tasks and features that would be covered in each, Using La TeX and word – Accessing, overview of toolbars, saving files, Using help and resources, rulers, format painter in word.

**Task 2:** Using LaTeX and Word to create a project certificate. Features to be covered:- Formatting Fonts in word, Drop Cap in word, Applying Text effects, Using Character Spacing, Borders and Colors, Inserting Header and Footer, Using Date and Time option in both LaTeX and Word.

**Task 3:** Creating project abstract Features to be covered:-Formatting Styles, Inserting table, Bullets and Numbering, Changing Text Direction, Cell alignment, Footnote, Hyperlink, Symbols, Spell Check, Track Changes.

**Task 4:** Creating a Newsletter: Features to be covered:- Table of Content, Newspaper columns, Images from files and clipart, Drawing toolbar and Word Art, Formatting Images, Textboxes, Paragraphs and Mail Merge in word.

### **EXCEL**

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

**Task 1:** Creating a Scheduler - Features to be covered: Gridlines, Format Cells, Summation, auto fill, Formatting Text

**Task 2:** Calculating GPA -. Features to be covered:- Cell Referencing, Formulae in excel – average, std. deviation, Charts, Renaming and Inserting worksheets, Hyper linking, Count function,

### **LOOKUP/VLOOKUP**

**Task 3:** Split cells, freeze panes, group and outline, Sorting, Boolean and logical operators, Conditional formatting

### **POWER POINT**

**Task 1:** Students will be working on basic power point utilities and tools which help them create basic power point presentations. PPT Orientation, Slide Layouts, Inserting Text, Word Art, Formatting Text, Bullets and Numbering, Auto Shapes, Lines and Arrows in PowerPoint.

**Task 2:** Interactive presentations - Hyperlinks, Inserting –Images, Clip Art, Audio, Video, Objects, Tables and Charts.

**Task 3:** Master Layouts (slide, template, and notes), Types of views (basic, presentation, slide slotter, notes etc), and Inserting – Background, textures, Design Templates, Hidden slides.

### **AI TOOLS – ChatGPT**

**Task 1:** Prompt Engineering: Experiment with different types of prompts to see how the model responds. Try asking questions, starting conversations, or even providing incomplete sentences to see how the model completes them.

• Ex: Prompt: "You are a knowledgeable AI. Please answer the following question: What is the capital of France?"

**Task 2:** Creative Writing: Use the model as a writing assistant. Provide the beginning of a story or a description of a scene, and let the model generate the rest of the content. This can be a fun way to brainstorm creative ideas

• Ex: Prompt: "In a world where gravity suddenly stopped working, people started floating upwards. Write a story about how society adapted to this new reality."

**Task 3:** Language Translation: Experiment with translation tasks by providing a sentence in one language and asking the model to translate it into another language. Compare the output to see how accurate and fluent the translations are.

• Ex:Prompt: "Translate the following English sentence to French: 'Hello, how are you doing today?'"

**Reference Books:**

1. Comdex Information Technology course tool kit, Vikas Gupta, WILEY Dream tech, 2003
2. The Complete Computer upgrade and repair book, Cheryl A Schmidt, WILEY Dream tech, 2013, 3rd edition
3. Introduction to Information Technology, ITL Education Solutions limited, Pearson Education, 2012, 2nd edition
4. PC Hardware - A Handbook, Kate J. Chase, PHI (Microsoft)
5. LaTeX Companion, Leslie Lamport, PHI/Pearson.
6. IT Essentials PC Hardware and Software Companion Guide, David Anfins on and Ken Quamme. – CISCO Press, Pearson Education, 3rd edition
7. IT Essentials PC Hardware and Software Labs and Study Guide, Patrick Regan– CISCO Press, Pearson Education, 3rd edition

B.Tech. (I Sem.)

23FE51-COMMUNICATIVE ENGLISH LAB

L	T	P	Cr.
0	0	2	1

**Pre-requisites:** Nil

**Course Objectives:** The main objective of introducing this course, Communicative English Laboratory, is to expose the students to a variety of self-instructional, learner friendly modes of language learning. The students will get trained in basic communication skills and also make them ready to face job interviews.

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

- CO1:** Understand the different aspect of the English language proficiency with emphasis on LSRW skills. (**Understand-L2**)
- CO2:** Apply Communication Skills through various language learning activities. (**Apply-L3**)
- CO3:** Identifying the English speech sounds, stress, rhythm, intonation and syllable division for better listening and speaking, comprehension. (**Understand-L2**)
- CO4:** Exhibit professionalism in participating in debates and group discussions. (**Apply-L3**)

**List of Topics:**

1. Vowels & Consonants
2. Neutralization/Accent Rules
3. Communication Skills & JAM
4. Role Play or Conversational Practice
5. E-mail Writing
6. Resume Writing, Cover letter, SOP
7. Group Discussions-methods & practice
8. Debates - Methods & Practice
9. PPT Presentations/ Poster Presentation
10. Interviews Skills

**Suggested Software:**

1. Walden Infotech
2. Young India Films

**Reference Books:**

1. Raman Meenakshi, Sangeeta-Sharma. Technical Communication. Oxford Press.2018.
2. Taylor Grant: English Conversation Practice, Tata McGraw-Hill Education India, 2016
3. Hewing's, Martin. Cambridge Academic English (B2). CUP, 2012.
4. J. Sethi & P.V. Dhamija. A Course in Phonetics and Spoken English, (2nd Ed) , Kindle, 2013

**Web Resources:**

**Spoken English:**

1. [www.esl-lab.com](http://www.esl-lab.com)
2. [www.englishmedialab.com](http://www.englishmedialab.com)
3. [www.englishinteractive.net](http://www.englishinteractive.net)
4. <https://www.britishcouncil.in/english/online>

5. <http://www.letstalkpodcast.com/>
6. [https://www.youtube.com/c/mmmEnglish\\_Emma/featured](https://www.youtube.com/c/mmmEnglish_Emma/featured)
7. <https://www.youtube.com/c/ArnelsEverydayEnglish/featured>
8. <https://www.youtube.com/c/engvidAdam/featured>
9. <https://www.youtube.com/c/EnglishClass101/featured>
10. <https://www.youtube.com/c/SpeakEnglishWithTiffani/playlists>
11. [https://www.youtube.com/channel/UCV1h\\_cBE0Drdx19qkTM0WNw](https://www.youtube.com/channel/UCV1h_cBE0Drdx19qkTM0WNw)

**Voice & Accent:**

1. <https://www.youtube.com/user/letstalkaccent/videos>
2. <https://www.youtube.com/c/EngLanguageClub/featured>
3. [https://www.youtube.com/channel/UC\\_OskgZBoS4dAnVUgJVexc](https://www.youtube.com/channel/UC_OskgZBoS4dAnVUgJVexc)
4. [https://www.youtube.com/channel/UCNfm92h83W2i2ijc5Xwp\\_IA](https://www.youtube.com/channel/UCNfm92h83W2i2ijc5Xwp_IA)

L	T	P	Cr.
0	0	2	1

**Pre-requisites:** Nil

**Course Objectives:**

- To enable the students to perform different types of volumetric titrations.
- To provides an overview of preparation of polymers, nano-materials and analytical techniques.

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

**CO1:** Distinguish different types of titrations in volumetric analysis after performing the experiments listed in the syllabus. **(Analyze-L4)**

**CO2:** Acquire practical knowledge related to preparation of bakelite and nano-materials. **(Apply-L3)**

**CO3:** Measure the strength of acid present in Pb-Acid battery. **(Apply-L3)**

**CO4:** Analyze important parameters of water to check its suitability for drinking purpose and industrial applications. **(Analyze-L4)**

**CO5:** Improve individual / teamwork skills, communication and report writing skills with ethical values.

**List of Experiments:**

- Measurement of 10Dq by spectro photometric method
- Conductometric titration of strong acid vs. strong base
- Conductometric titration of weak acid vs. strong base
- Determination of cell constant and conductance of solutions
- Potentiometry - determination of redox potentials and emfs
- Determination of Strength of an acid in Pb-Acid battery
- Preparation of a Bakelite
- Verify Lambert-Beer's law
- Wavelength measurement of sample through UV-Visible Spectroscopy
- Identification of simple organic compounds by IR
- Preparation of nano materials by precipitation method
- Estimation of Ferrous Iron by Dichrometry

**Reference:**

"Vogel's Quantitative Chemical Analysis 6th Edition 6th Edition" Pearson Publications by J. Mendham, R.C. Denney, J.D. Barnes and B. Sivasankar

L	T	P	Cr.
0	0	3	1.5

**Pre-requisite:** Mathematics, Basic Computer Terminology

**Course Objective:** The course aims to give students hands – on experience and train them on the concepts of the C- programming language.

**Course Outcomes (CO):** At the end of this course, the student will be able to:

**CO1:** Read, understand, and trace the execution of programs written in C language.  
(Understand-L2)

**CO2:** Apply the right control structure for solving the problem. (Apply-L3)

**CO3:** Develop, Debug and Execute programs to demonstrate the applications of arrays, functions, pointers and files in C. (Apply-L3)

**CO4:** Improve individual / teamwork skills, communication and report writing skills with ethical values.

## UNIT I

### WEEK 1

**Objective:** Getting familiar with the programming environment on the computer and writing the first program.

**Suggested Experiments/Activities:**

**Tutorial 1:** Problem-solving using Computers.

**Lab1:** Familiarization with programming environment

- i) Basic Linux environment and its editors like Vi, Vim & Emacs etc.
- ii) Exposure to Turbo C, gcc
- iii) Writing simple programs using printf(), scanf()

### WEEK 2

**Objective:** Getting familiar with how to formally describe a solution to a problem in a series of finite steps both using textual notation and graphic notation.

**Suggested Experiments /Activities:**

**Tutorial 2:** Problem-solving using Algorithms and Flow charts.

**Lab 2:** Converting algorithms/flow charts into C Source code.

Developing the algorithms/flowcharts for the following sample programs

- i) Sum and average of 3 numbers
- ii) Conversion of Fahrenheit to Celsius and vice versa
- iii) Simple interest calculation

### WEEK 3

**Objective:** Learn how to define variables with the desired data-type, initialize them with appropriate values and how arithmetic operators can be used with variables and constants.

**Suggested Experiments/Activities:**

**Tutorial 3:** Variable types and type conversions:

**Lab 3:** Simple computational problems using arithmetic expressions.

- i) Finding the square root of a given number
- ii) Finding compound interest
- iii) Area of a triangle using heron's formulae
- iv) Distance travelled by an object

**UNIT II****WEEK 4**

**Objective:** Explore the full scope of expressions, type-compatibility of variables & constants and operators used in the expression and how operator precedence works.

**Suggested Experiments/Activities:**

**Tutorial 4:** Operators and the precedence and as associativity:

**Lab 4:** Simple computational problems using the operator' precedence and associativity

- i) Evaluate the following expressions.
  - a.  $A+B*C+(D*E) + F*G$
  - b.  $A/B*C-B+A*D/3$
  - c.  $A+++B---A$
  - d.  $J=(i++) + (++i)$
- ii) Find the maximum of three numbers using conditional operator
- iii) Take marks of 5 subjects in integers, and find the total, average in float

**WEEK 5**

**Objective:** Explore the full scope of different variants of “if construct” namely if-else, null else, if-else if\*-else, switch and nested-if including in what scenario each one of them can be used and how to use them. Explore all relational and logical operators while writing conditionals for “if construct”.

**Suggested Experiments/Activities:**

**Tutorial 5:** Branching and logical expressions:

**Lab 5:** Problems involving if-then-else structures.

- i) Write a C program to find the max and min of four numbers using if-else.
- ii) Write a C program to generate electricity bill.
- iii) Find the roots of the quadratic equation.
- iv) Write a C program to simulate a calculator using switch case.
- v) Write a C program to find the given year is a leap year or not.

**WEEK 6**

**Objective:** Explore the full scope of iterative constructs namely while loop, do-while loop and for loop in addition to structured jump constructs like break and continue including when each of these statements is more appropriate to use.

**Suggested Experiments/Activities:**

**Tutorial 6:** Loops, while and for loops

**Lab 6:** Iterative problems e.g., the sum of series

- i) Find the factorial of given number using any loop.
- ii) Find the given number is a prime or not.
- iii) Compute sine and cos series
- iv) Checking a number palindrome
- v) Construct a pyramid of numbers.

**UNIT III****WEEK 7:**

**Objective:** Explore the full scope of Arrays construct namely defining and initializing 1-D and 2-D and more generically n-D arrays and referencing individual array elements from the defined array. Using integer 1-D arrays, explore search solution linear search.

**Suggested Experiments/Activities:**

**Tutorial 7:** 1 D Arrays: searching.

**Lab 7:** 1D Array manipulation, linear search

- i) Find the min and max of a 1-D integer array.
- ii) Perform linear search on 1D array.
- iii) The reverse of a 1D integer array
- iv) Find 2's complement of the given binary number.
- v) Eliminate duplicate elements in an array.

**WEEK 8:**

**Objective:** Explore the difference between other arrays and character arrays that can be used as Strings by using null character and get comfortable with string by doing experiments that will reverse a string and concatenate two strings. Explore sorting solution bubble sort using integer arrays.

**Suggested Experiments/Activities:**

**Tutorial 8:** 2 D arrays, sorting and Strings.

**Lab 8:** Matrix problems, String operations, Bubble sort

- i) Addition of two matrices
- ii) Multiplication two matrices
- iii) Sort array elements using bubble sort
- iv) Concatenate two strings without built-in functions
- v) Reverse a string using built-in and without built-in string functions

**UNIT IV****WEEK 9:**

**Objective:** Explore pointers to manage a dynamic array of integers, including memory allocation and value initialization, resizing changing and reordering the contents of an array and memory de-allocation using malloc(), calloc(), realloc() and free() functions. Gain experience processing command-line arguments received by C

**Suggested Experiments/Activities:**

**Tutorial 9:** Pointers, structures and dynamic memory allocation

**Lab 9:** Pointers and structures, memory dereference.

- i) Write a C program to find the sum of a 1D array using malloc()
- ii) Write a C program to find the total, average of n students using structures
- iii) Enter n students data using calloc() and display failed students list
- iv) Read student name and marks from the command line and display the student details along with the total.
- v) Write a C program to implement realloc()

**WEEK 10:**

**Objective:** Experiment with C Structures, Unions, bit fields and self-referential structures (Singly linked lists) and nested structures

**Suggested Experiments/Activities:**

**Tutorial 10:** Bitfields, Self-Referential Structures, Linked lists

**Lab10 :** Bitfields, linked lists

- i) Read and print a date using dd/mm/yyyy format using bit-fields and differentiate the same without using bit- fields
- ii) Create and display a singly linked list using self-referential structure.
- iii) Demonstrate the differences between structures and unions using a C program.
- iv) Write a C program to shift/rotate using bitfields.
- v) Write a C program to copy one structure variable to another structure of the same type.

**UNIT V****WEEK 11:**

**Objective:** Explore the Functions, sub-routines, scope and extent of variables, doing some experiments by parameter passing using call by value. Basic methods of numerical integration

**Suggested Experiments/Activities:**

**Tutorial 11:** Functions, call by value, scope and extent,

**Lab 11:** Simple functions using call by value, solving differential equations using Euler's theorem.

- i) Write a C function to calculate NCR value.
- ii) Write a C function to find the length of a string.
- iii) Write a C function to transpose of a matrix.

- iv) Write a C function to demonstrate numerical integration of differential equations using Euler's method

**WEEK 12:**

**Objective:** Explore how recursive solutions can be programmed by writing recursive functions that can be invoked from the main by programming at-least five distinct problems that have naturally recursive solutions.

**Suggested Experiments/Activities:**

**Tutorial 12:** Recursion, the structure of recursive calls

**Lab 12:** Recursive functions

- i) Write a recursive function to generate Fibonacci series.
- ii) Write a recursive function to find the LCM of two numbers.
- iii) Write a recursive function to find the factorial of a number.
- iv) Write a C Program to implement Ackermann function using recursion.
- v) Write a recursive function to find the sum of series.

**WEEK 13:**

**Objective:** Explore the basic difference between normal and pointer variables, Arithmetic operations using pointers and passing variables to functions using pointers

**Suggested Experiments/Activities:**

**Tutorial 13:** Call by reference, dangling pointers

**Lab 13:** Simple functions using Call by reference, Dangling pointers.

- i) Write a C program to swap two numbers using call by reference.
- ii) Demonstrate Dangling pointer problem using a C program.
- iii) Write a C program to copy one string into another using pointer.
- iv) Write a C program to find no of lowercase, uppercase, digits and other characters using pointers.

**WEEK14:**

**Objective:** To understand data files and file handling with various file I/O functions. Explore the differences between text and binary files.

**Suggested Experiments/Activities:**

**Tutorial 14:** File handling

**Lab 14:** File operations

- i) Write a C program to write and read text into a file.
- ii) Write a C program to write and read text into a binary file using fread() and fwrite()
- iii) Copy the contents of one file to another file.
- iv) Write a C program to merge two files into the third file using command-line arguments.
- v) Find no. of lines, words and characters in a file
- vi) Write a C program to print last n characters of a given file.

**Textbooks:**

1. Ajay Mittal, Programming in C: A practical approach, Pearson.
2. Byron Gottfried, Schaum's Outline of Programming with C, McGraw Hill

**Reference Books:**

1. Brian W. Kernighan and Dennis M. Ritchie, The C Programming Language, Prentice Hall of India
2. C Programming, A Problem-Solving Approach, Forouzan, Gilberg, Prasad, CENGAGE

B.Tech. (II Sem.)

23FE04-ENGINEERING PHYSICS

L	T	P	Cr.
3	0	0	3

**Course Objectives:**

To bridge the gap between the Physics in school at 10+2 level and UG level engineering courses by identifying the importance of the optical phenomenon like interference, diffraction etc, enlightening the periodic arrangement of atoms in crystalline solids and concepts of quantum mechanics, introduce novel concepts of dielectric and magnetic materials, physics of semiconductors.

**Course Outcomes:**

- CO1:** Analyse the intensity variation of light due to interference, diffraction, and Polarization (Apply-L3)
- CO2:** Understand the basics of crystals and their structures (Understand-L2)
- CO3:** Summarize various types of polarization of dielectrics and classify the magnetic materials (Understand-L2)
- CO4:** Explain fundamentals of quantum mechanics and free electron theory of metals (Understand-L2)
- CO5:** Identify the type of semiconductor using Hall Effect (Apply-L3)

**UNIT I****Wave Optics**

**Interference:** Introduction - Principle of superposition –Interference of light - Interference in thin films (Reflection Geometry) & applications - Colours in thin films- Newton’s Rings, Determination of wavelength and refractive index.

**Diffraction:** Introduction - Fresnel and Fraunhofer diffractions - 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****Crystallography and X-ray diffraction**

**Crystallography:** Space lattice, Basis, Unit Cell and lattice parameters – Bravais Lattices – crystal systems (3D) – coordination number - packing fraction of SC, BCC & FCC – Miller indices – separation between successive (hkl) planes.

**X-ray diffraction:** Bragg’s law - X-ray Diffractometer – crystal structure determination by Laue’s and powder methods

**UNIT III****Dielectric and Magnetic Materials**

**Dielectric Materials:** Introduction - Dielectric polarization - Dielectric polarizability, Susceptibility, Dielectric constant and Displacement Vector – Relation between the electric vectors - Types of polarizations- Electronic (Quantitative), Ionic (Quantitative) and Orientation polarizations (Qualitative) - Lorentz internal field - Clausius- Mossotti equation – complex dielectric constant – Frequency dependence of polarization – dielectric loss

**Magnetic Materials:** Introduction - Magnetic dipole moment - Magnetization-Magnetic susceptibility and permeability – Atomic origin of magnetism - 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.

#### UNIT IV

##### Quantum Mechanics and Free electron Theory

**Quantum Mechanics:** 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.

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

#### UNIT V

##### Semiconductors

**Semiconductors:** Formation of energy bands – classification of crystalline solids – 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 and its applications.

##### Textbooks:

1. A Text book of Engineering Physics, M. N. Avadhanulu, P.G.Kshirsagar & TVS Arun Murthy, S. Chand Publications, 11th Edition 2019.
2. Engineering Physics - D.K.Bhattacharya and Poonam Tandon, Oxford press (2015)

##### Reference Books:

1. Engineering Physics - B.K. Pandey and S. Chaturvedi, Cengage Learning 2021.
2. Engineering Physics - Shatendra Sharma, Jyotsna Sharma, Pearson Education, 2018.
3. Engineering Physics” - Sanjay D. Jain, D. Sahasrabudhe and Girish, University Press. 2010
4. Engineering Physics - M.R. Srinivasan, New Age international publishers (2009).

**Web Resources:** <https://www.loc.gov/rr/scitech/selected-internet/physics.html>

L	T	P	Cr.
3	0	0	3

**Pre-requisites:** Nil

**Course Objectives:**

- To enlighten the learners in the concept of differential equations and multivariable calculus.
- 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

- CO1** Solve the differential equations related to various engineering fields (Unit – I&II)  
(Apply-L3)
- CO2** Apply knowledge of partial differentiation in modelling and solving of Partial differential equations. (Apply-L3)
- CO3** Interpret the physical meaning of different operators such as gradient, curl and divergence. (Apply-L3)
- CO4** Evaluate the work done against a field, circulation and flux using Vector Calculus.  
(Apply-L3)

#### UNIT I

##### **Differential equations of first order and first degree**

Linear differential equations – Bernoulli's equations- Exact equations and equations reducible to exact form. Applications: Newton's Law of cooling – Law of natural growth and decay Electrical circuits.

#### UNIT II

##### **Linear differential equations of higher order (Constant Coefficients)**

Definitions, homogenous and non-homogenous, complimentary function, general solution, particular integral, Wronskian, Method of variation of parameters. Simultaneous linear equations, Applications to L-C-R Circuit problems and Simple Harmonic motion.

#### UNIT III

##### **Partial Differential Equations**

Introduction and formation of Partial Differential Equations by elimination of arbitrary constants and arbitrary functions, solutions of first order linear equations using Lagrange's method. Homogeneous Linear Partial differential equations with constant coefficients.

#### UNIT IV

##### **Vector differentiation**

Scalar and vector point functions, vector operator Del, Del applies to scalar point functions Gradient, Directional derivative, del applied to vector point functions-Divergence and Curl, vector identities.

#### UNIT V

##### **Vector integration**

LWithoutegral-circulation-work done, surface integral-flux, Green's theorem in the plane (without proof), Stoke's theorem (without proof), volume integral, Divergence theorem (without proof) and related problems.

**Textbooks:**

1. Higher Engineering Mathematics, B. S. Grewal, Khanna Publishers, 2017, 44th Edition
2. Advanced Engineering Mathematics, Erwin Kreyszig, John Wiley & Sons, 2018, 10<sup>th</sup> Edition.

**Reference Books:**

1. Thomas Calculus, George B. Thomas, Maurice D. Weir and Joel Hass, Pearson Publishers, 2018, 14th Edition.
2. Advanced Engineering Mathematics, Dennis G. Zill and Warren S. Wright, Jones and Bartlett, 2018.
3. Advanced Modern Engineering Mathematics, Glyn James, Pearson publishers, 2018, 5th Edition.
4. Advanced Engineering Mathematics, R. K. Jain and S. R. K. Iyengar, Alpha Science International Ltd., 2021 5th Edition (9th reprint).
5. Higher Engineering Mathematics, B. V. Ramana, , McGraw Hill Education, 2017

**23EE01-BASIC ELECTRICAL & ELECTRONICS  
ENGINEERING**

**B.Tech. (II Sem.)**

L	T	P	Cr.
3	0	0	3

**Course Objectives**

To expose to the field of electrical & electronics engineering, laws and principles of electrical/ electronic engineering and to acquire fundamental knowledge in the relevant field.

**Course Outcomes:** After the completion of the course students will be able to

**CO1:** Extract electrical variables of AC & DC circuits using fundamental laws. **(Understand-L2)**

**CO2:** Understand the operation of electrical machines and measuring instruments. **(Understand-L2)**

**CO3:** Classify various energy resources, safety measures and interpret electricity bill generation in electrical systems. **(Understand-L2)**

**PART A: BASIC ELECTRICAL ENGINEERING**

**UNIT I**

**DC & AC Circuits**

**DC Circuits:** Electrical circuit elements (R, L and C), Ohm's Law and its limitations, KCL & KVL, series, parallel, series-parallel circuits, Super Position theorem, Simple numerical problems.

**AC Circuits:** A.C. Fundamentals: Equation of AC Voltage and current, waveform, time period, frequency, amplitude, phase, phase difference, average value, RMS value, form factor, peak factor, Voltage and current relationship with phasor diagrams in R, L, and C circuits, Concept of Impedance, Active power, reactive power and apparent power, Concept of power factor (Simple Numerical problems).

**UNIT II**

**Machines and Measuring Instruments**

**Machines:** Construction, principle and operation of (i) DC Motor, (ii) DC Generator, (iii) Single Phase Transformer, (iv) Three Phase Induction Motor and (v) Alternator, Applications of electrical machines.

**Measuring Instruments:** Construction and working principle of Permanent Magnet Moving Coil (PMMC), Moving Iron (MI) Instruments and Wheat Stone Bridge.

**UNIT III**

**Energy Resources, Electricity Bill & Safety Measures**

**Energy Resources:** Conventional and non-conventional energy resources; Layout and operation of various Power Generation systems: Hydel, Nuclear, Solar & Wind power generation.

**Electricity bill:** Power rating of household appliances including air conditioners, PCs, Laptops, Printers, etc. Definition of "unit" used for consumption of electrical energy, two-part electricity tariff, calculation of electricity bill for domestic consumers.

**Equipment Safety Measures:** Working principle of Fuse and Miniature circuit breaker (MCB), merits and demerits. Personal safety measures: Electric Shock, Earthing and its types, Safety Precautions to avoid shock.

**Textbooks:**

1. Basic Electrical Engineering, D. C. Kulshreshtha, Tata McGraw Hill, 2019, First Edition
2. Power System Engineering, P.V. Gupta, M.L. Soni, U.S. Bhatnagar and A. Chakrabarti, Dhanpat Rai & Co, 2013
3. Fundamentals of Electrical Engineering, Rajendra Prasad, PHI publishers, 2014, Third Edition

**Reference Books:**

1. Basic Electrical Engineering, D. P. Kothari and I. J. Nagrath, Mc Graw Hill, 2019, Fourth Edition
2. Principles of Power Systems, V.K. Mehtha, S.Chand Technical Publishers, 2020

3. Basic Electrical Engineering, T. K. Nagsarkar and M. S. Sukhija, Oxford University Press, 2017
4. Basic Electrical and Electronics Engineering, S. K. Bhattacharya, Person Publications, 2018, Second Edition.

**Web Resources:**

1. <https://nptel.ac.in/courses/108105053>
2. <https://nptel.ac.in/courses/108108076>

**PART B: BASIC ELECTRONICS ENGINEERING**

**Course Objectives:** To teach the fundamentals of semiconductor devices and its applications, principles of digital electronics.

**Course Outcomes:** After the completion of the course students will be able to

**CO4:** Interpret the characteristics of various semiconductor devices (**Understand-L2**)

**CO5:** Infer the operation of rectifiers, amplifiers. (**Understand-L2**)

**CO6:** Contrast various logic gates, sequential and combinational logic circuits. (**Understand-L2**)

**UNIT I**

**SEMICONDUCTOR DEVICES**

Introduction - Evolution of electronics – Vacuum tubes to nano electronics - Characteristics of PN Junction Diode — Zener Effect — Zener Diode and its Characteristics. Bipolar Junction Transistor — CB, CE, CC Configurations and Characteristics — Elementary Treatment of Small Signal CE Amplifier.

**UNIT II**

**BASIC ELECTRONIC CIRCUITS AND INSTRUMENTATION**

**Rectifiers and power supplies:** Block diagram description of a dc power supply, working of a full wave bridge rectifier, capacitor filter (no analysis), working of simple zener voltage regulator. **Amplifiers:** Block diagram of Public Address system, Circuit diagram and working of common emitter (RC coupled) amplifier with its frequency response. **Electronic Instrumentation:** Block diagram of an electronic instrumentation system.

**UNIT III**

**DIGITAL ELECTRONICS**

Overview of Number Systems, Logic gates including Universal Gates, BCD codes, Excess-3 code, Gray code, Hamming code. Boolean Algebra, Basic Theorems and properties of Boolean Algebra, Truth Tables and Functionality of Logic Gates – NOT, OR, AND, NOR, NAND, XOR and XNOR. Simple combinational circuits–Half and Full Adders. Introduction to sequential circuits, Flip flops, Registers and counters (Elementary Treatment only)

**Textbooks:**

1. R. L. Boylestad & Louis Nashlesky, Electronic Devices & Circuit Theory, Pearson Education, 2021.
2. R. P. Jain, Modern Digital Electronics, 4th Edition, Tata Mc Graw Hill, 2009

**Reference Books:**

1. R. S. Sedha, A Textbook of Electronic Devices and Circuits, S. Chand & Co, 2010.
2. Santiram Kal, Basic Electronics- Devices, Circuits and IT Fundamentals, Prentice Hall, India, 2002.
3. R. T. Paynter, Introductory Electronic Devices & Circuits – Conventional Flow Version, Pearson Education, 2009.

L	T	P	Cr.
2	0	2	3

**Pre-requisites:** Nil

**Course Objectives:**

- To enable the students with various concepts like dimensioning, conventions and standards related to Engineering Drawing
- To impart knowledge on the projection of points, lines and plane surfaces
- To improve the visualization skills for better understanding of projection of solids
- To develop the imaginative skills of the students required to understand Section of solids and Developments of surfaces.
- To make the students understand the viewing perception of a solid object in Isometric and Perspective projections.

**Course Outcomes:**

- CO1:** Understand the principles of engineering drawing, including engineering curves, scales, orthographic and isometric projections. **(Understand- L2)**
- CO2:** Draw and interpret orthographic projections of points, lines, planes and solids in front, top and side views.**(Apply- L3)**
- CO3:** Understand and draw projection of solids in various positions in first quadrant. **(Apply-L3)**
- CO4:** Able to draw the development of surfaces of simple objects. **(Apply- L3)**
- CO5:** Prepare isometric and orthographic sections of simple solids **(Apply- L3)**

**UNIT I**

**Introduction:** Lines, Lettering and Dimensioning, Geometrical Constructions and Constructing regular polygons by general methods. **(Covered theoretically in class. Not for the end examination)**

**Scales:** Plain scales, diagonal scales and vernier scales. **(Covered theoretically in class. Not for the end examination)**

**Curves:** construction of ellipse, parabola and hyperbola by general, Cycloids, Involutives, Normal and tangent to Curves.

**Orthographic Projections:** Reference plane, importance of reference lines or Plane, Projections of a point situated in any one of the four quadrants

**UNIT II.**

**Projections of Straight Lines:** Projections of straight lines parallel to both reference planes, perpendicular to one reference plane and parallel to other reference plane, inclined to one reference plane and parallel to the other reference plane. Projections of Straight Line Inclined to both the reference planes

**Projections of Planes:** regular planes Perpendicular to both reference planes, parallel to one reference plane and inclined to the other reference plane; plane inclined to both the reference planes.

**UNIT III**

**Projections of Solids:** Types of solids: Polyhedra and Solids of revolution. Projections of solids in simple positions: Axis perpendicular to horizontal plane, Axis perpendicular to vertical plane and Axis parallel to both the reference planes, Projection of Solids with axis inclined to one reference plane and parallel to another plane.

#### UNIT IV

**Sections of Solids:** Perpendicular and inclined section planes, Sectional views and True shape of section, Sections of solids in simple position only.

**Development of Surfaces:** Methods of Development: Parallel line development and radial line development. Development of a cube, prism, cylinder, pyramid and cone.

#### UNIT V

**Conversion of Views:** Conversion of isometric views to orthographic views; Conversion of orthographic views to isometric views.

**Computer graphics:** Creating 2D&3D drawings of objects including PCB and Transformations using Auto CAD (**Not for end examination**).

#### Textbook:

N. D. Bhatt, Engineering Drawing, Charotar Publishing House, 2016.

#### Reference Books:

1. Engineering Drawing, K.L. Narayana and P. Kannaiah, Tata McGraw Hill, 2013.
2. Engineering Drawing, M.B.Shah and B.C. Rana, Pearson Education Inc,2009.
3. Engineering Drawing with an Introduction to AutoCAD, Dhananjay Jolhe, Tata McGraw Hill, 2017.

L	T	P	Cr.
3	0	0	3

**Course Objectives:**

- To provide the knowledge of basic data structures and their implementations.
- To understand importance of data structures in context of writing efficient programs.
- To develop skills to apply appropriate data structures in problem solving.

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

**CO1:** Understand the role of linear and non- linear data structures in organizing and accessing data (**Understand-L2**)

**CO2:** Implement abstract data type (ADT) and data structures for given application. (**Apply-L3**)

**CO3:** Design algorithms based on techniques like linked list, stack, queue, trees etc. (**Apply-L3**)

**CO4:** Apply the appropriate linear and nonlinear data structure techniques for solving a problem. (**Apply-L3**)

**CO5:** Design hash-based solutions for specific problems. (**Apply-L3**)

**UNIT I**

**Introduction to Linear Data Structures:** Definition and importance of linear data structures, Abstract data types (ADTs) and their implementation, Overview of time and space complexity analysis for linear data structures. **Searching Techniques:** Linear & Binary Search, **Sorting Techniques:** Bubble sort, Selection sort, Insertion Sort

**UNIT II**

**Linked Lists:** Singly linked lists: representation and operations, doubly linked lists and circular linked lists, Comparing arrays and linked lists, Applications of linked lists.

**UNIT III**

**Stacks:** Introduction to stacks: properties and operations, implementing stacks using arrays and linked lists, Applications of stacks in expression evaluation, backtracking, reversing list etc.

**UNIT IV**

**Queues:** Introduction to queues: properties and operations, implementing queues using arrays and linked lists, Applications of queues in breadth-first search, scheduling, etc.

**Deque:** Introduction to deque (double-ended queues), Operations on deque and their applications.

**UNIT V**

**Trees:** Introduction to Trees, Binary Search Tree – Insertion, Deletion & Traversal

**Hashing:** Brief introduction to hashing and hash functions, Collision resolution techniques: chaining and open addressing, Hash tables: basic implementation and operations, Applications of hashing in unique identifier generation, caching, etc.

**Textbooks:**

1. Data Structures and algorithm analysis in C, Mark Allen Weiss, Pearson, 2nd Edition.
2. Fundamentals of data structures in C, Ellis Horowitz, Sartaj Sahni, Susan AndersonFreed, Silicon Press, 2008

**Reference Books:**

1. Algorithms and Data Structures: The Basic Toolbox by Kurt Mehlhorn and Peter Sanders
2. C Data Structures and Algorithms by Alfred V. Aho, Jeffrey D. Ullman, and John E. Hopcroft
3. Problem Solving with Algorithms and Data Structures" by Brad Miller and David Ranum
4. Introduction to Algorithms by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein
5. Algorithms in C, Parts 1-5 (Bundle): Fundamentals, Data Structures, Sorting, Searching, and Graph Algorithms" by Robert Sedgewick

L	T	P	Cr.
0	0	2	1

**Course Objectives:**

To study the concepts of optical phenomenon like interference, diffraction etc., recognize the importance of energy gap in the study of conductivity and Hall effect in semiconductors and study the parameters and applications of dielectric and magnetic materials by conducting experiments.

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

**CO1:** Analyze the wave properties of light using optical instruments (**Apply-L3**).

**CO2:** Estimate the elastic moduli of various materials and acceleration due to gravity (**Apply-L3**).

**CO3:** Demonstrate the vibrations in stretched strings (**Understand-L2**).

**CO4:** Evaluate dielectric constant and magnetic field of circular coil carrying current (**Apply-L3**).

**CO5:** Examine the characteristics of semiconductor devices (**Apply-L3**).

**List of Experiments:**

1. Determination of radius of curvature of a given Plano-convex lens by Newton's rings.
2. Determination of wavelengths of different spectral lines in mercury spectrum using diffraction grating in normal incidence configuration.
3. Verification of Brewster's law
4. Determination of dielectric constant using charging and discharging method.
5. Study the variation of B versus H by magnetizing the magnetic material (B-H curve).
6. Determination of wavelength of Laser light using diffraction grating.
7. Estimation of Planck's constant using photoelectric effect.
8. Determination of the resistivity of semiconductors by four probe methods.
9. Determination of energy gap of a semiconductor using p-n junction diode.
10. Magnetic field along the axis of a current carrying circular coil by Stewart Gee's Method.
11. Determination of Hall voltage and Hall coefficient of a given semiconductor using Hall effect.
12. Determination of temperature coefficients of a thermistor.
13. Determination of acceleration due to gravity and radius of Gyration by using a compound pendulum.
14. Determination of magnetic susceptibility by Kundt's tube method.
15. Determination of rigidity modulus of the material of the given wire using Torsional pendulum.

16. Sonometer: Verification of laws of stretched string.

17. Determination of young's modulus for the given material of wooden scale by nonuniform bending (or double cantilever) method.

18. Determination of Frequency of electrically maintained tuning fork by Melde's experiment.

**Note: Any TEN of the listed experiments are to be conducted. Out of which any TWO experiments may be conducted in virtual mode.**

**References:**

A Textbook of Practical Physics - S. Balasubramanian, M.N. Srinivasan, S. Chand Publishers, 2017.

**Web Resources**

- [www.vlab.co.in](http://www.vlab.co.in)
- <https://phet.colorado.edu/en/simulations/filter?subjects=physics&type=html,prototype>



**Note: Minimum Six Experiments to be performed.**

### **PART B: ELECTRONICS ENGINEERING LAB**

#### **Course Objectives:**

To impart knowledge on the principles of digital electronics and fundamentals of electron devices & its applications.

#### **List of Experiments:**

1. Plot V-I characteristics of PN Junction diode A) Forward bias B) Reverse bias.
2. Plot V – I characteristics of Zener Diode and its application as voltage Regulator.
3. Implementation of half wave and full wave rectifiers
4. Plot Input & Output characteristics of BJT in CE and CB configurations
5. Frequency response of CE amplifier.
6. Simulation of RC coupled amplifier with the design supplied
7. Verification of Truth Table of AND, OR, NOT, NAND, NOR, Ex-OR, Ex-NOR gates using ICs.
8. Verification of Truth Tables of S-R, J-K& D flip flops using respective ICs.

**Tools / Equipment Required:** DC Power supplies, Multi meters, DC Ammeters, DC Voltmeters, AC Voltmeters, CROs, all the required active devices.

#### **References:**

1. R. L. Boylestad & Louis Nashlesky, Electronic Devices & Circuit Theory, Pearson Education, 2021.
2. R. P. Jain, Modern Digital Electronics, 4th Edition, Tata Mc Graw Hill, 2009
3. R. T. Paynter, Introductory Electronic Devices & Circuits – Conventional Flow Version, Pearson Education, 2009.

**Note: Minimum Six Experiments to be performed. All the experiments shall be implemented using both Hardware and Software.**

B.Tech. (II Sem.)

23ME51-ENGINEERING WORKSHOP

L	T	P	Cr.
0	0	3	1.5

**Pre-requisites** : Nil

**Course Objectives:**

To familiarize students with wood working, sheet metal operations, fitting, electrical house wiring skills, and basic repairs of two-wheeler vehicle.

**Course Outcomes:**

**CO1:** Identify workshop tools and their operational capabilities. (**Remember-L1**)

**CO2:** Practice on manufacturing of components using workshop trades including fitting, carpentry, foundry, and welding. (**Understand-L2**)

**CO3:** Modal various basic prototypes in fitting trade. (**Apply-L3**)

**CO4:** Apply basic electrical engineering knowledge for House Wiring Practice (**Apply-L3**)

**SYLLABUS**

1. Demonstration: Safety practices and precautions to be observed in workshop.
2. Wood Working: Familiarity with different types of woods and tools used in wood working and make following joints.
  - a) Half – Lap joint b) Mortise and Tenon joint c) Corner Dovetail joint or Bridle joint
3. Sheet Metal Working: Familiarity with different types of tools used in sheet metal working, Developments of following sheet metal job from GI sheets.
  - a) Tapered tray b) Conical funnel c) Elbow pipe d) Brazing
4. Fitting: Familiarity with different types of tools used in fitting and do the following fitting exercises.
  - a) V-fit b) Dovetail fit c) Semi-circular fit d) Bicycle tire puncture and change of two-wheeler tyre
5. Electrical Wiring: Familiarity with different types of basic electrical circuits and make the following connections.
  - a) Parallel and series b) Two-way switch c) Godown lighting d) Tube light e) Three phase motor f) Soldering of wires
6. Foundry Trade: Demonstration and practice on Moulding tools and processes, Preparation of Green Sand Moulds for given Patterns.
7. Welding Shop: Demonstration and practice on Arc Welding and Gas welding. Preparation of Lap joint and Butt joint.
8. Plumbing: Demonstration and practice of Plumbing tools, Preparation of Pipe joints with coupling for same diameter and with reducer for different diameters.

9. Basic repairs of Two-wheeler vehicle – Demonstration of working of two-wheeler vehicle and its repairs.

**Textbooks:**

1. Basic Workshop Technology: Manufacturing Process, Felix W.; Independently Published, 2019. Workshop Processes, Practices and Materials; Bruce J. Black, Routledge publishers, 5th Edn. 2015.
2. A Course in Workshop Technology Vol I. & II, B.S. Raghuwanshi, Dhanpath Rai & Co., 2015 & 2017.

**Reference Books:**

1. Elements of Workshop Technology, Vol. I by S. K. Hajra Choudhury & Others, Media Promoters and Publishers, Mumbai. 2007, 14th edition
2. Workshop Practice by H. S. Bawa, Tata-McGraw Hill, 2004.
3. Wiring Estimating, Costing and Contracting; Soni P.M. & Upadhyay P.A.; Atul Prakashan, 2021-22.

L	T	P	Cr.
0	0	3	1.5

**Pre-requisites** : **Computer Programming Lab**

**Course Objectives:**

The course aims to strengthen the ability of the students to identify and apply the suitable data structure for the given real-world problem. It enables them to gain knowledge in practical applications of data structures.

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

**CO1:** Apply Linear Data Structures for organizing the data efficiently (**Apply-L3**)

**CO2:** Apply Non-Linear Data Structures to organize data efficiently (**Apply-L3**)

**CO3:** Develop and implement hashing techniques for solving problems. (**Apply-L3**)

**CO4:** Improve individual / teamwork skills, communication and report writing skills with ethical values.

**List of Experiments:**

**Exercise 1: Array Manipulation**

- i) Write a program to reverse an array.
- ii) C Programs to implement the Searching Techniques – Linear & Binary Search
- iii) C Programs to implement Sorting Techniques –Bubble, Selection and Insertion sort

**Exercise 2: Linked List Implementation**

- i) Implement a singly linked list and perform insertion and deletion operations.
- ii) Develop a program to reverse a linked list iteratively and recursively.
- iii) Solve problems involving linked list traversal and manipulation.

**Exercise 3: Linked List Applications**

- i) Create a program to detect and remove duplicates from a linked list.
- ii) Implement a linked list to represent polynomials and perform addition.
- iii) Implement a double-ended queue (deque) with essential operations.

**Exercise 4: Double Linked List Implementation**

- i) Implement a doubly linked list and perform various operations to understand its properties and applications.
- ii) Implement a circular linked list and perform insertion, deletion, and traversal.

**Exercise 5: Stack Operations**

- i) Implement a stack using arrays and linked lists.
- ii) Write a program to evaluate a postfix expression using a stack.
- iii) Implement a program to check for balanced parentheses using a stack.

**Exercise 6: Queue Operations**

- i) Implement a queue using arrays and linked lists.
- ii) Develop a program to simulate a simple printer queue system.

iii) Solve problems involving circular queues.

**Exercise 7: Stack and Queue Applications**

- i) Use a stack to evaluate an infix expression and convert it to postfix.
- ii) Create a program to determine whether a given string is a palindrome or not.
- iii) Implement a stack or queue to perform comparison and check for symmetry.

**Exercise 8: Binary Search Tree**

- i) Implementing a BST using Linked List.
- ii) Traversing of BST.

**Exercise 9: Hashing**

- i) Implement a hash table with collision resolution techniques.
- ii) Write a program to implement a simple cache using hashing.

**Textbooks:**

1. Data Structures and algorithm analysis in C, Mark Allen Weiss, Pearson, 2nd Edition.
2. Fundamentals of data structures in C, Ellis Horowitz, Sartaj Sahni, Susan AndersonFreed, Silicon Press, 2008

**Reference Books:**

1. Algorithms and Data Structures: The Basic Toolbox by Kurt Mehlhorn and Peter Sanders
2. C Data Structures and Algorithms by Alfred V. Aho, Jeffrey D. Ullman, and John E. Hopcroft
3. Problem Solving with Algorithms and Data Structures" by Brad Miller and David Ranum
4. Introduction to Algorithms by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein
5. Algorithms in C, Parts 1-5 (Bundle): Fundamentals, Data Structures, Sorting, Searching, and Graph Algorithms by Robert Sedgewick

B.Tech. (III Sem.)

**23FE11-DISCRETE MATHEMATICS AND  
GRAPH THEORY**

L	T	P	Cr.
3	0	0	3

**Prerequisites:** Mathematics courses of first year of study.

**Course Objectives:**

- 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:**

- CO1:** Construct mathematical arguments using logical connectives and quantifiers and verify them. **(Apply- L3)**
- CO2:** Demonstrate the basic terminology of functions, relations, lattices and their operations. **(Understand -L2)**
- CO3:** Illustrate the basic principles/techniques to solve different combinatorial problems and linear recurrence relations. **(Apply- L3)**
- CO4:** Demonstrate the different types of graphs. **(Understand -L2)**
- CO5:** Apply the properties of graphs to solve the graph theory problems in computer science. **(Apply- L3)**

**UNIT-I: Mathematical Logic:**

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

**UNIT-II: Set Theory:**

**Sets:** Operations on Sets, Principle of Inclusion-Exclusion, Relations: Properties, Operations, Partition and Covering, Transitive Closure, Equivalence, Compatibility and Partial Ordering, Hasse Diagrams, Functions: Bijective, Composition, Inverse, Permutation, and Recursive Functions, Lattice and its Properties.

**UNIT-III: Combinatorics and Recurrence Relations:**

Basis of Counting, Permutations, Permutations with Repetitions, Circular and Restricted Permutations, Combinations, Restricted Combinations, Binomial and Multinomial Coefficients and Theorems.

**Recurrence Relations:**

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

**UNIT-IV: Graph Theory:**

Basic Concepts, Graph Theory and its Applications, Subgraphs, Graph Representations: Adjacency and Incidence Matrices, Isomorphic Graphs, Paths and Circuits, Eulerian and Hamiltonian Graphs,

**Unit-V: Multi Graphs**

Multigraphs, Bipartite and Planar Graphs, Euler's Theorem, Graph Colouring and Covering, Chromatic Number, Spanning Trees, Prim's and Kruskal's Algorithms, BFS and DFS Spanning Trees.

**TEXT BOOKS:**

1. Discrete Mathematical Structures with Applications to Computer Science, J. P. Tremblay and P. Manohar, Tata McGraw Hill.
2. Elements of Discrete Mathematics-A Computer Oriented Approach, C. L.Liu and D. P. Mohapatra, 3rd Edition, Tata McGraw Hill.
3. Theory and Problems of Discrete Mathematics, Schaum's Outline Series, Seymour Lipschutz and Marc Lars Lipson, 3rd Edition, McGraw Hill.

**REFERENCE BOOKS:**

1. Discrete Mathematics for Computer Scientists and Mathematicians, J. L.Mott, A. Kandel and T. P. Baker, 2nd Edition, Prentice Hall of India.
2. Discrete Mathematical Structures, Bernand Kolman, Robert C. Busby and Sharon Cutler Ross, PHI.
3. Discrete Mathematics, S. K. Chakraborty and B.K. Sarkar, Oxford, 2011.
4. Discrete Mathematics and its Applications with Combinatorics and Graph Theory, K. H. Rosen, 7th Edition, Tata McGraw Hill.

**23HS01-UHV2 - UNDERSTANDING HARMONY AND  
ETHICAL HUMAN CONDUCT**

L	T	P	Cr.
2	1	0	3

B.Tech. (III Sem.)

**Course Objectives:**

- To help the students appreciate the essential complementary between 'VALUES' and 'SKILLS' to ensure sustained happiness and prosperity which are the core aspirations of all human beings.
- To facilitate the development of a Holistic perspective among students towards life and profession as well as towards happiness and prosperity based on a correct understanding of the Human reality and the rest of existence. Such holistic perspective forms the basis of Universal Human Values and movement towards value-based living in a natural way.
- To highlight plausible implications of such a Holistic understanding in terms of ethical human conduct, trustful and mutually fulfilling human behavior and mutually enriching interaction with Nature.

**Course Outcomes:** At the end of the course student able to

- CO1:** Describe the terms like Natural Acceptance, Happiness and Prosperity (L2)  
**CO2:** Identify one's self, and one's surroundings (family, society nature) (L2)  
**CO3:** Relate human values with human relationship and human society. (L2)  
**CO4:** Illustrate the need for universal human values and harmonious existence (L2)  
**CO5:** Develop as socially and ecologically responsible engineers (L3)

**Course Topics:**

The course has 28 lectures and 14 tutorials in 5 modules. The lectures and tutorials are of 1- hour duration. Tutorial sessions are to be used to explore and practice what has been proposed during the lecture sessions.

The Teacher's Manual provides the outline for lectures as well as practice sessions. The teacher is expected to present the issues to be discussed as propositions and encourage the students to have a dialogue.

**UNIT I Introduction to Value Education (6 lectures and 3 tutorials for practice session)**

**Lecture 1:** Right Understanding, Relationship and Physical Facility (Holistic Development and the Role of Education)

**Lecture 2:** Understanding Value Education

**Tutorial 1:** Practice Session PS1 Sharing about Oneself

**Lecture 3:** self-exploration as the Process for Value Education

**Lecture 4:** Continuous Happiness and Prosperity – the Basic Human Aspirations

**Tutorial 2:** Practice Session PS2 Exploring Human Consciousness

**Lecture 5:** Happiness and Prosperity – Current Scenario

**Lecture 6:** Method to Fulfill the Basic Human Aspirations

**Tutorial 3:** Practice Session PS3 Exploring Natural Acceptance

**UNIT II Harmony in the Human Being (6 lectures and 3 tutorials for practice session)**

**Lecture 7:** Understanding Human being as the Co-existence of the self and the body.

**Lecture 8:** Distinguishing between the Needs of the self and the body

**Tutorial 4:** Practice Session PS4 Exploring the difference of Needs of self and body.

**Lecture 9:** The body as an Instrument of the self

**Lecture 10:** Understanding Harmony in the self

**Tutorial 5:** Practice Session PS5 Exploring Sources of Imagination in the self

**Lecture 11:** Harmony of the self with the body

**Lecture 12:** Programme to ensure self-regulation and Health

**Tutorial 6:** Practice Session PS6 Exploring Harmony of self with the body

**UNIT III: Harmony in the Family and Society (6 lectures and 3 tutorials for practice session)**

**Lecture 13:** Harmony in the Family – the Basic Unit of Human Interaction

**Lecture 14:** 'Trust' – the Foundational Value in Relationship

**Tutorial 7:** Practice Session PS7 Exploring the Feeling of Trust

**Lecture 15:** 'Respect' – as the Right Evaluation

**Tutorial 8:** Practice Session PS8 Exploring the Feeling of Respect

**Lecture 16:** Other Feelings, Justice in Human-to-Human Relationship

**Lecture 17:** Understanding Harmony in the Society

**Lecture 18:** Vision for the Universal Human Order

**Tutorial 9:** Practice Session PS9 Exploring Systems to fulfil Human Goal

**UNIT IV: Harmony in the Nature/Existence (4 lectures and 2 tutorials for practice session)**

**Lecture 19:** Understanding Harmony in the Nature

**Lecture 20:** Interconnectedness, self-regulation and Mutual Fulfilment among the Four Orders of Nature

**Tutorial 10:** Practice Session PS10 Exploring the Four Orders of Nature

**Lecture 21:** Realizing Existence as Co-existence at All Levels

**Lecture 22:** The Holistic Perception of Harmony in Existence

**Tutorial 11:** Practice Session PS11 Exploring Co-existence in Existence.

**UNIT V: Implications of the Holistic Understanding – a Look at Professional Ethics (6 lectures and 3 tutorials for practice session)**

**Lecture 23:** Natural Acceptance of Human Values

**Lecture 24:** Definitiveness of (Ethical) Human Conduct

**Tutorial 12:** Practice Session PS12 Exploring Ethical Human Conduct

**Lecture 25:** A Basis for Humanistic Education, Humanistic Constitution and Universal Human Order

**Lecture 26:** Competence in Professional Ethics

**Tutorial 13:** Practice Session PS13 Exploring Humanistic Models in Education

**Lecture 27:** Holistic Technologies, Production Systems and Management Models-Typical Case Studies

**Lecture 28:** Strategies for Transition towards Value-based Life and Profession

**Tutorial 14:** Practice Session PS14 Exploring Steps of Transition towards Universal Human Order

**Practice Sessions for UNIT I – Introduction to Value Education**

PS1 Sharing about Oneself

PS2 Exploring Human Consciousness

PS3 Exploring Natural Acceptance

**Practice Sessions for UNIT II – Harmony in the Human Being**

PS4 Exploring the difference of Needs of self and body

PS5 Exploring Sources of Imagination in the self

PS6 Exploring Harmony of self with the body

**Practice Sessions for UNIT III – Harmony in the Family and Society**

PS7 Exploring the Feeling of Trust

PS8 Exploring the Feeling of Respect

PS9 Exploring Systems to fulfil Human Goal

**Practice Sessions for UNIT IV – Harmony in the Nature (Existence)**

PS10 Exploring the Four Orders of Nature

PS11 Exploring Co-existence in Existence

**Practice Sessions for UNIT V – Implications of the Holistic Understanding – a Look at**

Professional Ethics

PS12 Exploring Ethical Human Conduct

PS13 Exploring Humanistic Models in Education

PS14 Exploring Steps of Transition towards Universal Human Order

**Textbook and Teachers Manual****The Textbook**

- R R Gaur, R Asthana, G P Bagaria, A Foundation Course in Human Values and Professional Ethics, 2nd Revised Edition, Excel Books, New Delhi, 2019. ISBN 978-93- 87034-47-1

**The Teacher's Manual**

- R R Gaur, R Asthana, G P Bagaria, Teachers' Manual for A Foundation Course in Human Values and Professional Ethics, 2nd Revised Edition, Excel Books, New Delhi, 2019. ISBN 978-93-87034- 53-2

**Reference Books**

1. Jeevan Vidya: Ek Parichaya, A Nagaraj, Jeevan Vidya Prakashan, Amarkantak, 1999.
2. Human Values, A.N. Tripathi, New Age Intl. Publishers, New Delhi, 2004.
3. The Story of Stuff (Book).
4. The Story of My Experiments with Truth - by Mohandas Karamchand Gandhi
5. Small is Beautiful - E. F Schumacher.
6. Slow is Beautiful - Cecile Andrews
7. Economy of Permanence - J C Kumarappa
8. Bharat Mein Angreji Raj – Pandit Sunderlal
9. Rediscovering India - by Dharampal
10. Hind Swaraj or Indian Home Rule - by Mohandas K. Gandhi
11. India Wins Freedom - Maulana Abdul Kalam Azad
12. Vivekananda - Romain Rolland (English)
13. Gandhi - Romain Rolland (English)

**Mode of Conduct:**

Lecture hours are to be used for interactive discussion, placing the proposals about the topics at hand and motivating students to reflect, explore and verify them.

Tutorial hours are to be used for practice sessions.

While analyzing and discussing the topic, the faculty mentor's role is in pointing to essential elements to help in sorting them out from the surface elements. In other words, help the students explore the important or critical elements.

In the discussions, particularly during practice sessions (tutorials), the mentor encourages the student to connect with one's own self and do self-observation, self-reflection and self-exploration.

Scenarios may be used to initiate discussion. The student is encouraged to take up "ordinary" situations rather than "extra-ordinary" situations. Such observations and their analyses are shared and discussed with other students and faculty mentor, in a group sitting.

Tutorials (experiments or practical) are important for the course. The difference is that the laboratory is everyday life, and practical are how you behave and work in real life. Depending on

the nature of topics, worksheets, home assignment and/or activity are included. The practice sessions (tutorials) would also provide support to a student in performing actions commensurate to his/her beliefs. It is intended that this would lead to development of commitment, namely behaving and working based on basic human values.

It is recommended that this content be placed before the student as it is, in the form of a basic foundation course, without including anything else or excluding any part of this content. Additional content may be offered in separate, higher courses. This course is to be taught by faculty from every teaching department, not exclusively by any one department.

Teacher preparation with a minimum exposure to at least one 8-day Faculty Development Program on Universal Human Values is deemed essential.

### Online Resources:

1. <https://fdp-si.aicte-india.org/UHV-II%20Class%20Notes%20&%20Handouts/UHV%20Handout%201-Introduction%20to%20Value%20Education.pdf>
2. <https://fdp-si.aicte-india.org/UHV-II%20Class%20Notes%20&%20Handouts/UHV%20Handout%202-Harmony%20in%20the%20Human%20Being.pdf>
3. <https://fdp-si.aicte-india.org/UHV-II%20Class%20Notes%20&%20Handouts/UHV%20Handout%203-Harmony%20in%20the%20Family.pdf>
4. <https://fdp-si.aicte-india.org/UHV%20I%20Teaching%20Material/D3-S2%20Respect%20July%202023.pdf>
5. <https://fdp-si.aicte-india.org/UHV-II%20Class%20Notes%20&%20Handouts/UHV%20Handout%205-Harmony%20in%20the%20Nature%20and%20Existence.pdf>
6. <https://fdp-si.aicte-india.org/download/FDPTeachingMaterial/3-days%20FDP-SI%20UHV%20Teaching%20Material/Day%203%20Handouts/UHV%20D%20D3-S2A%20Und%20Nature-Existence.pdf>
7. <https://fdp-si.aicte-india.org/UHV%20II%20Teaching%20Material/UHV%20II%20Lecture%202023-25%20Ethics%20v1.pdf>
8. <https://www.studocu.com/in/document/kiet-group-of-institutions/universal-human-values/chapter-5-holistic-understanding-of-harmony-on-professional-ethics/62490385>  
[https://onlinecourses.swayam2.ac.in/aic22\\_ge23/preview](https://onlinecourses.swayam2.ac.in/aic22_ge23/preview)

**23IT01- DIGITAL LOGIC & COMPUTER  
ORGANIZATION**

L	T	P	Cr.
3	0	0	3

**B.Tech. (III Sem.)**

**Prerequisites:**

**Course Objectives:** The main objectives of the course is to

1. Provide students with a comprehensive understanding of digital logic design principles and computer organization fundamentals.
2. Describe memory hierarchy concepts.
3. Explain input/output (I/O) systems and their interaction with the CPU, memory, and peripheral devices.

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

- CO1:** Evaluate digital number systems and use Boolean algebra theorems, Properties and Canonical forms for digital logic circuit design. **(Understand- L2)**
- CO2:** Design Sequential logic circuits and understand basic functional blocks a computer system **(Apply- L3)**
- CO3:** Understand computer architecture and Data representation to perform computer arithmetic operations and processor organization. **(Understand- L2)**
- CO4:** Analyze the memory hierarchy in a computer system. **(Understand- L2)** **CO5:** Understand the I/O operations and the interfaces **(Understand-L2)**

**UNIT – I:**

**Data Representation:** Binary Numbers Fixed Point Representation. Floating Point Representation. Number base conversions, Octal and Hexadecimal Numbers, components, Signed binary numbers, Binary codes

**Digital Logic Circuits-I:** Basic Logic Functions, Logic gates, universal logic gates, Minimization of Logic expressions. K-Map Simplification, Combinational Circuits, Decoders, Multiplexers

**UNIT – II:**

**Digital Logic Circuits-II:** Sequential Circuits, Flip-Flops, Binary counters, Registers, Shift Registers, Ripple counters

**Basic Structure of Computers:** Computer Types, Functional units, Basic operational concepts, Bus structures, Software, Performance, multiprocessors and multi computers, Computer Generations, Von- Neumann Architecture

**UNIT – III:**

**Computer Arithmetic:** Addition and Subtraction of Signed Numbers, Design of Fast Adders, Multiplication of Positive Numbers, Signed-operand Multiplication, Fast Multiplication, Integer Division, Floating-Point Numbers and Operations

**Processor Organization:** Fundamental Concepts, Execution of a Complete Instruction, Multiple-Bus Organization, Hardwired Control and Multi programmed Control

**UNIT – IV:**

**The Memory Organization:** Basic Concepts, Semiconductor RAM Memories, Read-Only Memories, Speed, Size and Cost, Cache Memory, Performance Metrics, Virtual Memory, Memory Management Requirements, Secondary Storage Memory.

**UNIT – V:**

**Input/output Organization:** Accessing I/O Devices and Interfaces, Data Transfer Techniques, Direct Memory Access, Buses, Interface Circuits, Standard I/O Interfaces.

**Textbooks:**

1. Computer Organization, Carl Hamacher, Zvonko Vranesic, Safwat Zaky, 6th edition, McGraw Hill
2. Digital Design, 6th Edition, M. Morris Mano, Pearson Education.
3. Computer Organization and Architecture, William Stallings, 11th Edition, Pearson.

**Reference Books:**

1. Computer Systems Architecture, M. Moris Mano, 3rd Edition, Pearson
2. Computer Organization and Design, David A. Paterson, John L. Hennessy, Elsevier
3. Fundamentals of Logic Design, Roth, 5th Edition, Thomson

**Online Learning Resources:**

<https://nptel.ac.in/courses/106/103/106103068/>

**23CS04-ADVANCED DATA STRUCTURES &  
ALGORITHM ANALYSIS**

B.Tech. (III Sem.)

L	T	P	Cr.
3	0	0	3

**Prerequisites:** Data Structures

**Course Objectives:** The main objectives of the course is to

1. Provide knowledge on advance data structures frequently used in Computer Science domain
2. Develop skills in algorithm design techniques popularly used
3. Understand the use of various data structures in the algorithm design

**Course Outcomes:** After successful completion of the course the students are able to

- CO1:** Identify the characteristics of an algorithm, analyze its time and space complexity and construct balanced binary trees. **(Apply-L3)**
- CO2:** Understand Heap structures and graph terminology to perform various operations on non-linear data structures. **(Understand-L2)**
- CO3:** Apply Divide and Conquer, Greedy algorithm and dynamic programming for solving problems. **(Apply - L3)**
- CO4:** Analyze the backtracking and branch-and-bound search methods on optimization problems **(Apply - L3)**
- CO5:** Summarize the importance of NP-Hard and its applications. **(Understand-L2)**

**UNIT – I:**

**Introduction:** Algorithm Analysis, Space and Time Complexity analysis, Asymptotic Notations.

**AVL Trees** – Creation, Insertion, Deletion operations and Applications

**B-Trees** – Creation, Insertion, Deletion operations and Applications

**UNIT – II:**

**Heap Trees (Priority Queues)** – Min and Max Heaps, Operations and Applications

**Graphs** – Terminology, Representations, Basic Search and Traversals, Connected Components and Biconnected Components, applications

**Divide and Conquer:** The General Method, Finding Max-Min, Quick Sort, Merge Sort, Strassen's matrix multiplication,

**UNIT – III:**

**Greedy Method:** General Method, Job Sequencing with deadlines, Knapsack Problem, Minimum cost spanning trees, Single Source Shortest Paths, Optimal Storage on tapes, Huffman coding.

**UNIT – IV:**

**Dynamic Programming:** General Method, All pairs shortest paths, Single Source Shortest Paths– General Weights (Bellman Ford Algorithm), Optimal Binary Search Trees, 0/1 Knapsack, String Editing, Travelling Salesperson problem.

**UNIT – V:**

**Backtracking:** General Method, n-Queens Problem, Sum of Subsets problem, Graph Coloring.

**Branch and Bound:** The General Method, 0/1 Knapsack Problem, Travelling Salesperson problem

**Introduction to Complexity classes:** P and NP Problems, NP-Complete Problems.

**Textbooks:**

1. Fundamentals of Data Structures in C++, Horowitz, Ellis; Sahni, Sartaj; Mehta, Dinesh, 2nd Edition Universities Press
2. Computer Algorithms in C++, Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran, 2nd Edition University Press

**Reference Books:**

1. Data Structures and program design in C, Robert Kruse, Pearson Education Asia
2. An introduction to Data Structures with applications, Trembley & Sorenson, McGraw Hill
3. The Art of Computer Programming, Vol.1: Fundamental Algorithms, Donald E Knuth, Addison-Wesley, 1997.
4. Data Structures using C & C++: Langsam, Augenstein & Tanenbaum, Pearson, 1995 Algorithms + Data Structures & Programs: N.Wirth, PHI
6. Fundamentals of Data Structures in C++: Horowitz Sahni & Mehta, Galgottia Pub.
7. Data structures in Java: Thomas Standish, Pearson Education Asia

**Online Learning Resources:**

1. [https://www.tutorialspoint.com/advanced\\_data\\_structures/index.asp](https://www.tutorialspoint.com/advanced_data_structures/index.asp)
2. <http://peterindia.net/Algorithms.html>
3. Abdul Bari, Introduction to Algorithms (youtube.com)

**23CS05-OBJECT ORIENTED PROGRAMMING  
THROUGH JAVA**

L	T	P	Cr.
3	0	0	3

B.Tech. (III Sem.)

**Prerequisites:** Introduction to Programming

**Course Objectives:** The learning objectives of this course are to:

1. Identify Java language components and how they work together in applications
2. Learn the fundamentals of object-oriented programming in Java, including defining classes, invoking methods, using class libraries.
3. Learn how to extend Java classes with inheritance and dynamic binding and how to use exception handling in Java applications
4. Understand how to design applications with threads in Java
5. Understand how to use Java APIs for program development

**Course Outcomes:** After successful completion of the course the students are able to

**CO1:** Identify the syntax and semantics of java programming language and basic concepts of Java.  
(Understand-L2)

**CO2:** Understand the basic concepts of object-oriented programming (Understand-L2)

**CO3:** Develop reusable programs using the concepts of inheritance, polymorphism, and interfaces.  
(Apply-L3)

**CO4:** Apply the concepts of packages, exception handling & I/O streams to develop secure, error free, and efficient applications (Apply-L3)

**CO5:** Design multithreaded and GUI based applications which mimic the real word scenarios.  
(Apply-L3)

### UNIT I

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

**Data Types, Variables:** Introduction, Data Types in Java, Declaration of Variables, Data Types, Type Casting, Scope of Variable Identifier, Literal Constants, Symbolic Constants, Formatted Output with printf() Method, Static Variables and Methods, Attribute Final,

**Introduction to Operators:** Precedence and Associativity of Operators, Assignment Operator ( = ), Basic Arithmetic Operators, Increment (++) and Decrement (- -) Operators, Ternary Operator, Relational Operators, Boolean Logical Operators, Bitwise Logical Operators.

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

### UNIT II

**Classes and Objects:** Introduction, Class Declaration and Modifiers, Class Members, Declaration of Class Objects, Assigning One Object to Another, Access Control for Class Members, Accessing Private Members of Class.

**Constructors and Methods:** Introduction, Defining Methods, Constructor Methods for Class, Overloaded Constructor Methods, Overloaded Methods, Nested Classes, Passing Arguments by Value and by Reference, Keyword this. Class Objects as Parameters in Methods, Access Control, Recursive Methods, Nesting of Methods, Attributes Final and Static.

**String Handling in Java:** Introduction, Interface Char Sequence, Class String, Methods for Extracting Characters from Strings, Comparison, Modifying, Searching; Class String Buffer.

### UNIT III

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

**Inheritance:** Introduction, Relationship between classes- Has-a, Is-a, Process of Inheritance, Types of Inheritances, Universal Super Class-Object Class, Inhibiting Inheritance of Class Using Final, Access Control and Inheritance, Multilevel Inheritance, Application of Keyword Super, Constructor Method and Inheritance, Method Overriding, Dynamic Method Dispatch, Abstract Classes.

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

### UNIT IV

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

**Exception Handling:** Introduction, Hierarchy of Standard Exception Classes, Keywords try, catch, throw, throws, and finally Blocks, Multiple Catch Clauses, Class Throwable, Unchecked Exceptions, Checked Exceptions, Generating user defined exception.

**Java I/O and File:** Java I/O API, standard I/O streams, types, Byte streams, Character streams, Scanner class, Files in Java (Text Book 2)

### UNIT V

**Multithreaded Programming:** Introduction, Need for Multiple Threads., Multithreaded Programming for Multi-core Processor, Thread Class, Main Thread - Creation of New Threads, Thread States, Thread Priorities, Synchronization, Inter-thread Communication- producer consumer problem.

**Java Collections:** Introduction, Purpose of Collection Framework, Hierarchy of collection Interfaces / classes, Methods defined in Collection Interface, Interface Iterator, Collection classes/Interfaces –List, Set, Map.

**Java FX GUI:** Overview of AWT & Swings API, limitations, Java FX Scene Builder, Java FX App Window Structure, displaying text and image, event handling, laying out nodes in scene graph, mouse events (Text Book 3)

### Text Books:

1. JAVA one step ahead, Anitha Seth, B.L.Juneja, Oxford.
2. Joy with JAVA, Fundamentals of Object-Oriented Programming, Debasis Samanta, Monalisa Sarma, Cambridge, 2023.
3. JAVA 9 for Programmers, Paul Deitel, Harvey Deitel, 4th Edition, Pearson.

**References Books:**

1. The complete Reference Java, 11th edition, Herbert Schildt, TMH
2. Introduction to Java programming, 7th Edition, Y Daniel Liang, Pearson

**Online Resources:**

1. <https://nptel.ac.in/courses/106/105/106105191/>
2. [https://infyspringboard.onwingspan.com/web/en/app/toc/lex\\_auth\\_012880464547618816347\\_shared/overview](https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_012880464547618816347_shared/overview)

**23CS53-ADVANCED DATA STRUCTURES &  
ALGORITHM ANALYSIS LAB**

B.Tech. (III Sem.)

L	T	P	Cr.
0	0	3	1.5

**Prerequisites:** Data Structures Lab

**Course Objectives:** The objectives of the course is to

- Acquire practical skills in constructing and managing Data structures
- Apply the popular algorithm design methods in problem-solving scenarios

**Course Outcomes:** After successful completion of the course the students are able to

**CO1:** Implement balanced binary trees, heaps and graph traversals using arrays and linked list.  
(Apply-L3)

**CO2:** Implement Various Sorting Techniques. (Apply - L3)

**CO3:** Implement optimization problems using greedy, dynamic programming, backtracking and branch-and-bound techniques. (Apply - L3)

**CO4:** Improve individual / teamwork skills, communication & report writing skills with ethical values.

**Experiments covering the Topics:**

1. Operations on AVL trees, B-Trees, Heap Trees
2. Graph Traversals
3. Sorting techniques
4. Minimum cost spanning trees
5. Shortest path algorithms
6. 0/1 Knapsack Problem
7. Travelling Salesperson problem
8. Optimal Binary Search Trees
9. N-Queens Problem
10. Job Sequencing

**Sample Programs:**

1. Implement AVL Tree operations using linked list.
2. Construct B-Tree an order of 5 with a set of 100 random elements stored in array. Implement searching, insertion and deletion operations.
3. Construct Min and Max Heap using arrays, delete any element and display the content of the Heap.
4. Implement BFT and DFT for given graph, when graph is represented by
  - a) Adjacency Matrix
  - b) Adjacency Lists
5. Write a program for finding the biconnected components in a given graph.
6. Write a program to find maximum and minimum element in array using Divide and conquer.
7. Implement Quick sort and Merge sort and observe the execution time for various input sizes (Average, Worst and Best cases).
8. Compare the performance of Single Source Shortest Paths using Greedy method when the graph is represented by adjacency matrix and adjacency lists.
9. Implement Job Sequencing with deadlines using Greedy strategy.
10. Write a program to solve 0/1 Knapsack problem Using Dynamic Programming.
11. Implement N-Queens Problem Using Backtracking.
12. Implement Travelling Sales Person problem using Branch and Bound approach.

**Reference Books:**

1. Fundamentals of Data Structures in C++, Horowitz Ellis, Sahni Sartaj, Mehta, Dinesh, 2ndEdition, Universities Press
2. Computer Algorithms/C++ Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran, 2ndEdition, University Press
3. Data Structures and program design in C, Robert Kruse, Pearson Education Asia
4. An introduction to Data Structures with applications, Trembley& Sorenson, McGraw Hill

**Online Learning Resources:**

1. <http://cse01-iiith.vlabs.ac.in/>
2. <http://peterindia.net/Algorithms.html>

**23CS54-OBJECT ORIENTED PROGRAMMING  
THROUGH JAVA LAB**

L	T	P	Cr.
0	0	3	1.5

B.Tech. (III Sem.)

**Prerequisites:** Computer Programming Lab

**Course Objectives:** The aim of this course is to

- Practice object-oriented programming in the Java programming language
- Implement Classes, Objects, Methods, Inheritance, Exception, Runtime Polymorphism, User defined Exception handling mechanism
- Illustrate inheritance, Exception handling mechanism, JDBC connectivity
- Construct Threads, Event Handling, implement packages, Java FX GUI

**Course Outcomes:** After successful completion of the course the students are able to

**CO1:** Implement basic concepts of the java programming language. **(Apply-L3)**

**CO2:** Implement object-oriented programming concepts and exception handling **(Apply- L3).**

**CO3:** Design multithreaded, database and GUI based applications. **(Apply-L3)**

**CO4:** Improve individual / teamwork skills, communication & report writing skills with ethical values.

**Experiments covering the Topics:**

- Object Oriented Programming fundamentals- data types, control structures
- Classes, methods, objects, Inheritance, polymorphism,
- Exception handling, Threads, Packages, Interfaces
- Files, I/O streams, Java FX GUI

**Sample Experiments:**

**Exercise – 1:**

- 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**

- 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
- c. Write a JAVA program using String Buffer to delete, remove character.

**Exercise - 3**

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

**Exercise - 4**

- 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 shape

**Exercise - 5**

- 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?
- c. Write a JAVA program that implements Runtime polymorphism

**Exercise - 6**

- a. Write a JAVA program that describes exception handling mechanism
- b. Write a JAVA program Illustrating Multiple catch clauses
- c. Write a JAVA program for creation of Java Built-in Exceptions
- d. Write a JAVA program for creation of User Defined Exception

**Exercise - 7**

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

**Exercise – 8**

- a. Write a JAVA program that import and use the user defined packages
- b. Without writing any code, build a GUI that display text in label and image in an ImageView (use JavaFX)
- c. Build a Tip Calculator app using several JavaFX components and learn how to respond to user interactions with the GUI

**Exercise-9:**

- a. Implement the programs using List Interface and its implemented classes.
- b. Implement the programs using Set Interface and its implemented classes.
- c. Implement the programs using Map Interface and its implemented classes.

L	T	P	Cr.
0	1	2	2

**Course Objectives:** The main objectives of the course are to

- Make use of HTML elements and their attributes for designing static web pages
- Build a web page by applying appropriate CSS styles to HTML elements
- Experiment with JavaScript to develop dynamic web pages and validate forms

**Course Outcomes:** After successful completion of the course the students are able to

**CO1:** Design static web pages by using HTML elements. **(Apply-L3)**

**CO2:** Develop a web page by applying appropriate CSS styles to HTML elements. **(Apply-L3)**

**CO3:** Develop dynamic web pages and validate forms using JavaScript. **(Apply-L3)**

**CO4:** Improve individual / teamwork skills, communication & report writing skills with ethical values

**Experiments covering the Topics:**

- Lists, Links and Images
- HTML Tables, Forms and Frames
- HTML 5 and Cascading Style Sheets, Types of CSS
- Selector forms
- CSS with Color, Background, Font, Text and CSS Box Model
- Applying JavaScript - internal and external, I/O, Type Conversion
- JavaScript Conditional Statements and Loops, Pre-defined and User-defined Objects
- JavaScript Functions and Events
- Node.js

**Sample Experiments:**

### 1. Lists, Links and Images

a. Write a HTML program, to explain the working of lists.

Note: It should have an ordered list, unordered list, nested lists and ordered list in an unordered list and definition lists.

b. Write a HTML program, to explain the working of hyperlinks using <a> tag and href, target Attributes.

c. Create a HTML document that has your image and your friend's image with a specific height and width. Also when clicked on the images it should navigate to their respective profiles.

d. Write a HTML program, in such a way that, rather than placing large images on a page, the preferred technique is to use thumbnails by setting the height and width parameters to something like to 100\*100 pixels. Each thumbnail image is also a link to a full sized version of the image. Create an image gallery using this technique

### 2. HTML Tables, Forms and Frames

a. Write a HTML program, to explain the working of tables. (use tags: <table>, <tr>, <th>, <td> and attributes: border, rowspan, colspan)

b. Write a HTML program, to explain the working of tables by preparing a timetable. (Note: Use <caption> tag to set the caption to the table & also use cell spacing, cell padding, border, rowspan, colspan etc.).

c. Write a HTML program, to explain the working of forms by designing Registration form. (Note: Include text field, password field, number field, date of birth field, checkboxes, radio buttons, list

boxes using <select>&<option> tags, <text area> and two buttons ie: submit and reset. Use tables to provide a better view).

d. Write a HTML program, to explain the working of frames, such that page is to be divided into 3 parts on either direction. (Note: first frame  image, second frame  paragraph, third frame  hyperlink. And also make sure of using “no frame” attribute such that frames to be fixed).

### 3. HTML 5 and Cascading Style Sheets, Types of CSS

- Write a HTML program, that makes use of <article>, <aside>, <figure>, <figcaption>, <footer>, <header>, <main>, <nav>, <section>, <div>, <span> tags.
- Write a HTML program, to embed audio and video into HTML web page.
- Write a program to apply different types (or levels of styles or style specification formats) - inline, internal, external styles to HTML elements. (identify selector, property and value).

### 4. Selector forms

- Write a program to apply different types of selector forms
  - Simple selector (element, id, class, group, universal)
  - Combinator selector (descendant, child, adjacent sibling, general sibling)
  - Pseudo-class selector
  - Pseudo-element selector
  - Attribute selector

### 5. CSS with Color, Background, Font, Text and CSS Box Model

- Write a program to demonstrate the various ways you can reference a color in CSS.
- Write a CSS rule that places a background image halfway down the page, tilting it horizontally. The image should remain in place when the user scrolls up or down.
- Write a program using the following terms related to CSS font and text:
  - font-size
  - font-weight
  - font-style
  - text-decoration
  - text-transformation
  - text-alignment
- Write a program, to explain the importance of CSS Box model using
  - Content
  - Border
  - Margin
  - padding

### 6. Applying JavaScript - internal and external, I/O, Type Conversion

- Write a program to embed internal and external JavaScript in a web page.
- Write a program to explain the different ways for displaying output.
- Write a program to explain the different ways for taking input.
- Create a webpage which uses prompt dialogue box to ask a voter for his name and age. Display the information in table format along with either the voter can vote or not

### 7. Java Script Pre-defined and User-defined Objects

- Write a program using document object properties and methods.
- Write a program using window object properties and methods.
- Write a program using array object properties and methods.
- Write a program using math object properties and methods.
- Write a program using string object properties and methods.
- Write a program using regex object properties and methods.
- Write a program using date object properties and methods.
- Write a program to explain user-defined object by using properties, methods, accessors, constructors and display.

### 8. Java Script Conditional Statements and Loops

- Write a program which asks the user to enter three integers, obtains the numbers from the user and outputs HTML text that displays the larger number followed by the words “LARGER

NUMBER” in an information message dialog. If the numbers are equal, output HTML text as “EQUAL NUMBERS”.

- b. Write a program to display week days using switch case.
- c. Write a program to print 1 to 10 numbers using for, while and do-while loops.
- d. Write a program to print data in object using for-in, for-each and for-of loops
- e. Develop a program to determine whether a given number is an ‘ARMSTRONG NUMBER’ or not. [Eg: 153 is an Armstrong number, since sum of the cube of the digits is equal to the number i.e.,  $1^3 + 5^3 + 3^3 = 153$ ]
- f. Write a program to display the denomination of the amount deposited in the bank in terms of 100’s, 50’s, 20’s, 10’s, 5’s, 2’s & 1’s. (Eg: If deposited amount is Rs.163, the output should be 1-100’s, 1-50’s, 1-10’s, 1-2’s & 1-1’s)

### 9. Java Script Functions and Events

- a. Design a appropriate function should be called to display
  - i. Factorial of that number
  - ii. Fibonacci series up to that number
  - iii. Prime numbers up to that number
  - iv. palindrome or not
- b. Design a HTML having a text box and four buttons named Factorial, Fibonacci, Prime, and Palindrome. When a button is pressed an appropriate function should be called to display
  - i. Factorial of that number
  - ii. Fibonacci series up to that number
  - iii. Prime numbers up to that number
  - iv. Is it palindrome or not
- c. Write a program to validate the following fields in a registration page
  - i. Name (start with alphabet and followed by alphanumeric and the length should not be less than 6 characters)
  - ii. Mobile (only numbers and length 10 digits)
  - iii. E-mail (should contain format like [xxxxxxx@xxxxxx.xxx](mailto:xxxxxxx@xxxxxx.xxx) )

### Text Books:

1. Programming the World Wide Web, 7th Edition, Robert W Sebesta, Pearson, 2013.
2. Web Programming with HTML5, CSS and JavaScript, John Dean, Jones & Bartlett Learning, 2019 (Chapters 1-11).
3. Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node, Vasan Subramanian, 2nd edition, A Press, O’Reilly.

### Web Links:

1. <https://www.w3schools.com/html>
2. <https://www.w3schools.com/css>
3. <https://www.w3schools.com/js/>
4. <https://www.w3schools.com/nodejs>

L	T	P	Cr.
2	0	0	2

**Course Objectives:**

The course aims to equip students with the ability to define objective and constraint functions in terms of design variables for optimization problems, including both single and multi-variable problems with and without constraints. It covers the application of linear programming techniques, including the use of slack and surplus variables in the Simplex method, and the formulation of transportation and assignment problems as linear programming problems. Additionally, the course presents nonlinear programming techniques for both unconstrained and constrained problems, including the use of exterior and interior penalty functions

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

**CO1:** State and formulate optimization problems, with and without constraints, using design variables from an engineering design problem. **(Remember-L1)**

**CO2:** Apply classical optimization techniques to minimize or maximize a multi-variable objective function, with or without constraints, and arrive at an optimal solution. **(Understand- L2)**

**CO3:** Apply and solve transportation and assignment problems using the Linear Programming Simplex method. **(Apply-L3)**

**CO4:** Apply gradient and non-gradient methods to nonlinear optimization problems, using interior or exterior penalty functions for constraints, to derive optimal solutions. **(Apply-L3)**

**CO5:** Formulate and apply Dynamic Programming techniques to problems such as inventory control, production planning, and engineering design, to reach a final optimal solution from the current optimal solution. **(Analyse-L4)**

**UNIT I:** Introduction and Classical Optimization Techniques: Statement of an Optimization problem, design vector, design constraints, constraint surface, objective function, objective function surfaces, and classification of Optimization problems. Classical Optimization Techniques: Single variable Optimization, multi variable Optimization without constraints, necessary and sufficient conditions for minimum/maximum, multivariable Optimization with equality constraints. Solution by method of Lagrange multipliers, multivariable Optimization with inequality constraints, Kuhn – Tucker conditions

**UNIT II:** Linear Programming : Standard form of a linear programming problem, geometry of linear programming problems, definitions and theorems, solution of a system of linear simultaneous equations, pivotal reduction of a general system of equations, motivation to the simplex method, simplex algorithm.

**UNIT III:** Transportation Problem: Finding initial basic feasible solution by north – west corner rule, least cost method and Vogel’s approximation method, testing for optimality of balanced transportation problems, Special cases in transportation problem.

**UNIT IV:** Nonlinear Programming: Unconstrained cases, one – dimensional minimization methods: Classification, Fibonacci method, Univariate method, steepest descent method. Constrained cases– Characteristics of a constrained problem, Classification, Basic approach of Penalty Function method, Basic approaches of Interior and Exterior penalty function methods.

**UNIT V: Dynamic Programming:** Dynamic programming multistage decision processes, types, concept of sub optimization and the principle of optimality, computational procedure in dynamic programming, examples illustrating the calculus method of solution, examples illustrating the tabular method of solution.

**Textbooks:**

1. “Engineering optimization: Theory and practice”, S. S.Rao, New Age International (P) Limited, 3rd edition, 1998.
2. “Introductory Operations Research”, H.S. Kasene& K.D. Kumar, Springer (India), Pvt.LTd.

**Reference Books:**

1. “Optimization Methods in Operations Research and systems Analysis”, by K.V. Mital and C. Mohan, New Age International (P) Limited, Publishers, 3rd edition, 1996.
2. Operations Research, Dr.S.D.Sharma, Kedarnath, Ramnath& Co

L	T	P	Cr.
3	0	0	3

B.Tech. (IV Sem.)

23FE10-PROBABILITY &amp; STATISTICS

**Course Objectives:**

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

**Course Outcomes:** Upon successful completion of this course, the student should be able to

**CO1:** Classify the concepts of data science and its importance (**Understand-L2**)

**CO2:** Interpret the association of characteristics and through correlation and regression tools (**Analyze-L4**)

**CO3:** Apply discrete and continuous probability distributions (**Apply-L3**)

**CO4:** Design the components of a classical hypothesis test (**Apply-L3**)

**CO5:** Infer the statistical inferential methods based on small and large sampling tests (**Analyze-L4**)

**Unit – I: Descriptive statistics and methods for data science:**

Data science – Statistics Introduction – Population vs Sample –Collection of data – primary and secondary data – Type of variable: dependent and independent Categorical and Continuous variables – Data visualization – Measures of Central tendency – Measures of Variability – Skewness – Kurtosis.

**UNIT – II: Correlation and Regression:**

Correlation – Correlation coefficient – Rank correlation.

Linear Regression: Straight line – Multiple Linear Regression - Regression coefficients and properties – Curvilinear Regression: Parabola – Exponential – Power curves.

**UNIT – III: Probability and Distributions:**

Probability– Conditional probability and Baye’s theorem – Random variables – Discrete and Continuous random variables – Distribution functions – Probability mass function, Probability density function and Cumulative distribution functions – 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) – Point and Interval estimations – Maximum error of estimate – Central limit theorem (without proof) – Estimation using  $t$ ,  $\chi^2$  and F-distributions.

**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 – Test of significance for large samples and Small Samples: Single and difference means – Single and two proportions – Student’s  $t$ - test, F-test,  $\chi^2$  -test.

**Text Books:**

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

**Reference Books:**

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

L	T	P	Cr.
3	0	0	3

**Prerequisites:** Computer Organization

**Course Objectives:**

The main objectives of the course is to make student

- Understand the basic concepts and principles of operating systems, including process management, memory management, file systems, and Protection
- Make use of process scheduling algorithms and synchronization techniques to achieve better performance of a computer system.
- Illustrate different conditions for deadlock and their possible solutions.

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

**CO1:** Understand the fundamental concepts, functions, and structures of operating systems, including their design, implementation, and the various types of system calls and services.

**(Understand-L2)**

**CO2:** Understand process concepts, multithreading models, and CPU scheduling algorithms to effectively manage operations on processes, inter-process communication, and threading issues in operating systems. **(Understand-L2)**

**CO3:** Analyze synchronization tools, deadlock handling methods to solve critical section problems and ensure efficient process synchronization in operating systems. **(Apply-L3)**

**CO4:** Analyze different memory management techniques paging and segmentation to understand their suitability for various memory allocation scenarios. **(Apply-L3)**

**CO5:** Apply knowledge of file system structures and protection mechanisms to design and implement secure file management systems. **(Apply-L3)**

## SYLLABUS

### UNIT - I

**Operating Systems Overview:** Introduction, operating system functions, operating systems operations, Computing environments, Free and Open-Source Operating Systems.

**System Structures:** Operating System Services, User and Operating-System Interface, system calls, Types of System Calls, system programs, operating system Design and Implementation, operating system structure, Building and Booting an Operating System, Operating system debugging.

### UNIT - II

**Processes:** Process Concept, Process scheduling, Operations on processes, Inter-process communication.

**Threads and Concurrency:** Multithreading models, Thread libraries, Threading issues.

**CPU Scheduling:** Basic concepts, Scheduling criteria, Scheduling algorithms, Multiple processor scheduling.

### UNIT – III

**Synchronization Tools:** The Critical Section Problem, Peterson’s Solution, Mutex Locks, Semaphores, Monitors, Classic problems of Synchronization.

**Deadlocks:** system Model, Deadlock characterization, Methods for handling Deadlocks, Deadlock prevention, Deadlock avoidance, Deadlock detection, Recovery from Deadlock.

#### UNIT - IV

**Memory-Management Strategies:** Introduction, Contiguous memory allocation, Paging, Structure of the Page Table, Swapping.

**Virtual Memory Management:** Introduction, Demand paging, Copy-on-write, Page replacement, Allocation of frames, Thrashing

**Storage Management:** Overview of Mass Storage Structure, HDD Scheduling.

#### UNIT - V

**File System:** File System Interface: File concept, Access methods, Directory Structure; File system Implementation: File-system structure, File-system Operations, Directory implementation, Allocation method, Free space management; File-System Internals: File-System Mounting, Partitions and Mounting, File Sharing.

Protection: Goals of protection, Principles of protection, Protection Rings, Domain of protection, Access matrix.

#### Text Books:

1. Operating System Concepts, Silberschatz A, Galvin P B, Gagne G, 10<sup>th</sup> Edition, Wiley, 2018.
2. Modern Operating Systems, Tanenbaum A S, 4<sup>th</sup> Edition, Pearson , 2016

#### Reference Books:

1. Operating Systems -Internals and Design Principles, Stallings W, 9<sup>th</sup> edition, Pearson, 2018
2. Operating Systems: A Concept Based Approach, D.M Dhamdhere, 3<sup>rd</sup> Edition, McGraw- Hill, 2013

#### Online Learning Resources:

1. <https://nptel.ac.in/courses/106/106/106106144/>
2. <http://peterindia.net/OperatingSystems.html>

L	T	P	Cr.
3	0	0	3

**B.Tech. (IV Sem.) 23CS03-DATABASE MANAGEMENT SYSTEMS**

**Prerequisites:** Data Structures

**Course Objectives:** The main objective of this course is

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

**Course Outcomes:** After successful completion of the course the students are able to

**CO1:** Understand the foundation of database management system and various data models.

**(Understand-L2)**

**CO2:** Identify relational model concepts, implement various constraints, perform SQL queries and DML operations. **(Understand-L2)**

**CO3:** Apply SQL queries, functions, and work with nested queries, grouping, joins, views, and set operations. **(Apply-L3)**

**CO4:** Apply various normalization techniques for efficient data handling. **(Apply-L3)**

**CO5:** Understand Transaction management, recovery & indexing techniques. **(Understand-L2)**

**UNIT I:**

**Introduction:** Database system, Characteristics (Database Vs File System), Database Users, 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:**

**Entity Relationship Model:** Introduction, Representation of entities, attributes, entity set, relationship, relationship set, constraints, sub classes, super class, inheritance, specialization, generalization using ER Diagrams.

**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, Relational Algebra, Relational Calculus.

**UNIT III:**

**SQL: BASIC SQL:** Simple Database schema, data types, table definitions (create, alter), different DML operations (insert, delete, update). SQL querying (select and project) using where clause, arithmetic & logical operations, SQL functions (Date and Time, Numeric, String conversion). 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 Lossless join and

dependency preserving decomposition, (1NF, 2NF and 3 NF), concept of surrogate key, Boyce-Codd normal form (BCNF), MVD, Fourth normal form(4NF), Fifth Normal Form (5NF).

**UNIT V:**

**Transaction Processing and Concurrency Control:** Transaction State, ACID properties, Concurrent Executions, Serializability, Recoverability, Implementation of Isolation, Testing for Serializability. Two-Phase Locking Techniques for concurrency control: Types of Locks, Time stamp-based locking.

**Introduction to Recovery Protocols:** Recovery Concepts, No-UNDO/REDO Recovery Based on Deferred Update, Recovery Techniques Based on Immediate Update, Shadow Paging, ARIES.

**Introduction to Indexing:** Hash based Indexing

**Text Books:**

1. Database Management Systems, 3rd edition, Raghurama Krishnan, Johannes Gehrke, TMH (For Chapters 2, 3, 4)
2. Database System Concepts, 5th edition, Silberschatz, Korth, Sudarsan, TMH (For Chapter 1 and Chapter 5)

**Reference Books:**

1. Introduction to Database Systems, 8th edition, C J Date, Pearson.
2. Database Management System, 6th edition, Ramez Elmasri, Shamkant B. Navathe, Pearson
3. Database Principles Fundamentals of Design Implementation and Management, Corlos Coronel, Steven Morris, Peter Robb, Cengage Learning.

**Web-Resources:**

1. <https://nptel.ac.in/courses/106/105/106105175/>
2. [https://infyspringboard.onwingspan.com/web/en/app/toc/lex\\_auth\\_01275806667282022456\\_shared/overview](https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01275806667282022456_shared/overview)

L	T	P	Cr.
3	0	0	3

**Course Objectives:** The objectives of this course are to introduce

- Software life cycle models, Software requirements and SRS document.
- Project Planning, quality control and ensuring good quality software.
- Software Testing strategies, use of CASE tools, Implementation issues, validation & verification procedures.

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

**CO1:** Understanding the Evolution in Software Development and Implementation of Modern Software Development Practices. **(Understand - L2)**

**CO2:** Understand Software Project Management and Requirements Analysis Techniques. **(Understand - L2)**

**CO3:** Demonstrate Effective Software Design and Agile Practices. **(Apply - L3)**

**CO4:** Apply Coding, Testing, and Quality Management Practices. **(Apply - L3)**

**CO5:** Apply the usage of CASE tools, Software Maintenance process and Software Reuse. **(Apply - L3)**

#### UNIT I:

**Introduction:** Evolution, Software development projects, Exploratory style of software developments, Emergence of software engineering, Notable changes in software development practices, Computer system engineering.

**Software Life Cycle Models:** Basic concepts, Waterfall model and its extensions, Rapid application development, Agile development model, Spiral model.

#### UNIT II:

**Software Project Management:** Software project management complexities, Responsibilities of a software project manager, Metrics for project size estimation, Project estimation techniques, Empirical Estimation techniques, COCOMO, Halstead's software science, risk management.

**Requirements Analysis And Specification:** Requirements gathering and analysis, Software Requirements Specification (SRS), Formal system specification, Axiomatic specification, Algebraic specification, Executable specification and 4GL.

#### UNIT III:

**Software Design:** Overview of the design process, How to characterize a good software design? Layered arrangement of modules, Cohesion and Coupling, approaches to software design.

**Agility:** Agility and the Cost of Change, Agile Process, Extreme Programming (XP), Other Agile Process Models, Tool Set for the Agile Process (Text Book 2)

**Function-Oriented Software Design:** Overview of SA/SD methodology, Structured analysis, Developing the DFD model of a system, Structured design, Detailed design, and Design Review.

**User Interface Design:** Characteristics of a good user interface, Basic concepts, Types of user interfaces, Fundamentals of component-based GUI development, and user interface design methodology.

#### UNIT IV:

**Coding And Testing:** Coding, Code review, Software documentation, Testing, Black-box testing, White-Box testing, Debugging, Program analysis tools, Integration testing, Testing object-oriented programs, Smoke testing, and Some general issues associated with testing.

**Software Reliability And Quality Management:** Software reliability. Statistical testing, Software quality, Software quality management system, ISO 9000.SEI Capability maturity model. Few other important quality standards, and Six Sigma.

**UNIT V:**

**Computer-Aided Software Engineering (Case):** CASE and its scope, CASE environment, CASE support in the software life cycle, other characteristics of CASE tools, Towards second generation CASE Tool, and Architecture of a CASE Environment.

**Software Maintenance:** Characteristics of software maintenance, Software reverse engineering, Software maintenance process models and Estimation of maintenance cost.

**Software Reuse:** reuse- definition, introduction, reason behind no reuse so far, Basic issues in any reuse program, A reuse approach, and Reuse at organization level.

**Text Books:**

1. Fundamentals of Software Engineering, Rajib Mall, 5th Edition, PHI.
2. Software Engineering A practitioner's Approach, Roger S. Pressman, 9th Edition, Mc- Graw Hill International Edition.

**Reference Books:**

1. Software Engineering, Ian Sommerville, 10thEdition, Pearson.
2. Software Engineering, Principles and Practices, Deepak Jain, Oxford University Press.

**e-Resources:**

1. <https://nptel.ac.in/courses/106/105/106105182/>
2. [https://infyspringboard.onwingspan.com/web/en/app/toc/lex\\_auth\\_01260589506387148827\\_shared/overview](https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01260589506387148827_shared/overview)
3. [https://infyspringboard.onwingspan.com/web/en/app/toc/lex\\_auth\\_013382690411003904735\\_shared/overview](https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013382690411003904735_shared/overview)

L	T	P	Cr.
0	0	3	1.5

**Course Objectives:**

The main objectives of the course are to

- Provide insights into system calls, file systems, semaphores,
- Develop and debug CPU Scheduling algorithms, page replacement algorithms, thread implementation
- Implement Bankers Algorithms to Avoid the Dead Lock
- acquire the generic software development skill through various stages of software life cycle
- generate test cases for software testing

**Course Outcomes (COs):**

**CO1: Implement and evaluate** fundamental operating system concepts through programming exercises, including UNIX commands, system calls, and simulations of CPU scheduling algorithms. **(L3)**

**CO2: Analyze** synchronization mechanisms (semaphores and monitors) and memory allocation algorithms (first-fit, worst-fit, best-fit) through writing concurrent programs using the pthreads library. **(L4)**

**CO3: Design, develop,** and test real-world software using methodologies, UML, and testing principles. **(L3)**

**CO4:** Improve individual/ Teamwork skills, communication and report writing skills with ethical values.

**Experiments covering the Topics:**

- UNIX fundamentals, commands & system calls
- CPU Scheduling algorithms, thread processing
- IPC, semaphores, monitors, deadlocks
- Page replacement algorithms, file allocation strategies
- Memory allocation strategies
- Software Requirement Specification, DFD, CFD
- Software estimation, UML diagrams, test case design

**Sample Experiments in Operating Systems:**

1. Practicing of Basic UNIX Commands.
2. Write programs using the following UNIX operating system calls fork, exec, getpid, exit, wait, close, stat, opendir and readdir
3. Simulate UNIX commands like cp, ls, grep, etc.,
4. Simulate the following CPU scheduling algorithms  
a) FCFS b) SJF c) Priority d) Round Robin
5. Control the number of ports opened by the operating system with  
a) Semaphore b) Monitors.
6. Write a program to illustrate concurrent execution of threads using pthreads library.
7. Write a program to solve producer-consumer problem using Semaphores.
8. Implement the following memory allocation methods for fixed partition  
a) First fit b) Worst fit c) Best fit
9. Simulate the following page replacement algorithms  
a) FIFO b) LRU c) LFU
10. Simulate Paging Technique of memory management.
11. Implement Bankers Algorithm for Dead Lock avoidance and prevention

12. Simulate the following file allocation strategies
  - a) Sequential
  - b) Indexed
  - c) Linked
13. Download and install nachos operating system and experiment with it

**Sample Experiments in Software Engineering:**

- 1) Perform the following, for the following experiments:
  - i. Do the Requirement Analysis and Prepare SRS
  - ii. Draw E-R diagrams, DFD, CFD and structured charts for the project.
    - a. Course Registration System
    - b. Students Marks Analyzing System
    - c. Online Ticket Reservation System
    - d. Stock Maintenance
- 2) Consider any application, using COCOMO model, estimate the effort.
- 3) Consider any application, Calculate effort using FP oriented estimation model.
- 4) Draw the UML Diagrams for the problem a, b, c, d.
- 5) Design the test cases for e-Commerce application (Flipcart, Amazon)
- 6) Design the test cases for a Mobile Application (Consider any example from Appstore)
- 7) Design and Implement ATM system through UML Diagrams.

**Reference Books:**

1. Operating System Concepts, Silberschatz A, Galvin P B, Gagne G, 10th Edition, Wiley, 2018.
2. Modern Operating Systems, Tanenbaum A S, 4th Edition, Pearson, 2016
3. Operating Systems -Internals and Design Principles, Stallings W, 9th edition, Pearson, 2018
4. Operating Systems: A Concept Based Approach, D.M Dhamdhare, 3rd Edition, McGraw-Hill, 2013

**Online Learning Resources:**

1. <https://www.cse.iitb.ac.in/~mythili/os/>
2. <http://peterindia.net/OperatingSystems.html>
3. [www.cs.washington.edu/~tom/nachos](http://www.cs.washington.edu/~tom/nachos)

**23CS56- DATABASE MANAGEMENT  
SYSTEMS LAB**

L	T	P	Cr.
0	0	3	1.5

**B.Tech. (IV Sem.)**

**Course Objectives:** This Course will enable students to

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

**Course Outcomes:** After successful completion of the course the students are able to

**CO1:** Implement SQL queries using DDL/DML commands. **(Apply-L3)**

**CO2:** Apply different Integrity constraints & Normalization techniques for effective database design. **(Apply-L3)**

**CO3:** Implement PL/SQL including procedures, functions, cursors and triggers. **(Apply-L3)**

**CO4:** Improve individual / teamwork skills, communication & report writing skills with ethical values.

**Experiments covering the topics:**

- DDL, DML, DCL commands
- Queries, nested queries, built-in functions,
- PL/SQL programming- control structures
- Procedures, Functions, Cursors, Triggers,
- Database connectivity- ODBC/JDBC

**Sample Experiments:**

1. Creation, altering and dropping of tables and inserting rows into a table (use constraints while creating tables) examples using SELECT command.
2. Queries (along with sub Queries) using ANY, ALL, IN, EXISTS, NOTEXISTS, UNION, INTERSET, Constraints. Example:- Select the roll number and name of the student who secured fourth rank in the class.
3. Queries using Aggregate functions (COUNT, SUM, AVG, MAX and MIN), GROUP BY, HAVING and Creation and dropping of Views.
4. Queries using Conversion functions (to\_char, to\_number and to\_date), string functions (Concatenation, lpad, rpad, ltrim, rtrim, lower, upper, initcap, length, substr and instr), date functions (Sysdate, next\_day, add\_months, last\_day, months\_between, least, greatest, trunc, round, to\_char, to\_date)
5.
  - i. Create a simple PL/SQL program which includes declaration section, executable section and exception –Handling section (Ex. Student marks can be selected from the table and printed for those who secured first class and an exception can be raised if no records were found)
  - ii. Insert data into student table and use COMMIT, ROLLBACK and SAVEPOINT in PL/SQL block.
6. Develop a program that includes the features NESTED IF, CASE and CASE expression. The program can be extended using the NULLIF and COALESCE functions.

7. Program development using WHILE LOOPS, numeric FOR LOOPS, nested loops using ERROR Handling, BUILT –IN Exceptions, USE defined Exceptions, RAISE- APPLICATION ERROR.
8. Programs development using creation of procedures, passing parameters IN and OUT of PROCEDURES.
9. Program development using creation of stored functions, invoke functions in SQL Statements and write complex functions.
10. Develop programs using features parameters in a CURSOR, FOR UPDATE CURSOR, WHERE CURRENT of clause and CURSOR variables.
11. Develop Programs using BEFORE and AFTER Triggers, Row and Statement Triggers and INSTEAD OF Triggers
12. Create a table and perform the search operation on table using indexing and non- indexing techniques.

### **Design Database for any one of the following Case Studies**

#### **Case Study1: Hospital Management System**

Aim: XYZ hospital is a multi-specialty hospital that includes a number of departments, rooms, doctors, nurses, compounders, and other staff working in the hospital. Patients having different kinds of ailments come to the hospital and get checkup done from the concerned doctors. If required they are admitted in the hospital and discharged after treatment. The aim of this cases study is to design and develop a database for the hospital to maintain the records of various departments, rooms, and doctors in the hospital. It also maintains records of the regular patients, patients admitted in the hospital, the checkup of patients done by the doctors, the patients that have been operated, and patients discharged from the hospital.

**Description:** In hospital, there are many departments like Orthopedic, Pathology, Emergency, Dental, Gynecology, Anesthetics, I.C.U., Blood Bank, Operation Theater, Laboratory, M.R.I., Neurology, Cardiology, Cancer Department, Corpse, etc. There is an OPD where patients come and get a card (that is, entry card of the patient) for check up from the concerned doctor. After making entry in the card, they go to the concerned doctor's room and the doctor checks up the irailments. According to the ailments, the doctor either prescribes medicine or admits the patient in the concerned department. The patient may choose either private or general room according tohis/her need. But before getting admission in the hospital, the patient has to fulfill certain formalities of the hospital like room charges, etc. After the treatment is completed, the doctor is charges the patient. Before discharging from the hospital, the patient again has to complete certain formalities of the hospital like balance charges, test charges, operation charges (if any), blood charges, doctors' charges, etc. Next we talk about the doctors of the hospital. There are two types of the doctors in the hospital, namely, regular doctors and call on doctors. Regular doctors are those doctors who come to the hospital daily. Calls on doctors are those doctors who are called by the hospital if the concerned doctor is not available.

#### **Table Description:**

Following are the tables along with constraints used in Hospital Management Database

**Constraint:** Identity number is unique for each doctor and the corresponding department should exist in DEPARTMENT table.

**1.DEPARTMENT:** This table consists of details about the various departments in the hospital. The information stored in this table includes department name, department location, and facilities available in that department.

**Constraint:** Department name will be unique for each department.

**2. ALL\_DOCTORS:** This table stores information about all the doctors working for the hospital and the departments they are associated with. Each doctor is given an identity number starting with DR or DC prefixes only.

**3.DOC\_REG:** This table stores details of regular doctors working in the hospital. Doctors are referred to by their doctor number. This table also stores personal details of doctors like name, qualification, address, phone number, salary, date of joining, etc. Constraint: Doctor's number entered should contain DR only as a prefix and must exist in ALL\_DOCTORS table.

**4.DOC\_ON\_CALL:** This table stores details of doctors called by hospital when additional doctors are required. Doctors are referred to by their doctor number. Other personal details like name, qualification, fees per call, payment due, address, phone number, etc., are also stored.

**Constraint:** Doctor's number entered should contain DC only as a prefix and must exist in ALL\_DOCTORS table.

**5.PAT\_ENTRY:** The record in this table is created when any patient arrives in the hospital for a checkup. When patient arrives, a patient number is generated which acts as a primary key. Other details like name, age, sex, address, city, phone number, entry date, name of the doctor referred to, diagnosis, and department name are also stored. After storing the necessary details patient is sent to the doctor for checkup.

**Constraint:** Patient number should begin with prefix PT. Sex should be M or F only. Doctor's name and department referred must exist.

**6.PAT\_CHKUP:** This table stores the details about the patients who get treatment from the doctor referred to. Details like patient number from patient entry table, doctor number, date of checkup, diagnosis, and treatment are stored. One more field status is used to indicate whether patient is admitted, referred for operation or is a regular patient to the hospital. If patient is admitted, further details are stored in PAT\_ADMIT table. If patient is referred for operation, the further details are stored in PAT\_OPR table and if patient is a regular patient to the hospital, the further details are stored in PAT\_REG table.

Constraint: Patient number should exist in PAT\_ENTRY table and it should be unique.

**7.PAT\_ADMIT:** When patient is admitted, his/her related details are stored in this table. Information stored includes patient number, advance payment, mode of payment, room number, department, date of admission, initial condition, diagnosis, treatment, number of the doctor under whom treatment is done, attend an tname, etc.

**Constraint:** Patient number should exist in PAT\_ENTRY table. Department, doctor number, room number must be valid.

**8.PAT\_DIS:** An entry is made in this table whenever a patient gets discharged from the hospital. Each entry includes details like patient number, treatment given, treatment advice, payment made, mode of payment, date of discharge, etc.

**Constraint:** Patient number should exist in PAT\_ENTRY table.

**9.PAT\_REG:** Details of regular patients are stored in this table. Information stored includes date of visit, diagnosis, treatment, medicine recommended, status of treatment, etc.

**Constraint:** Patient number should exist in patient entry table. There can be multiple entries of one patient as patient might be visiting hospital repeatedly for checkup and there will be a try for patient's each visit.

**10.PAT\_OPR:** If patient is operated in the hospital, his/her details are stored in this table. Information stored includes patient number, date of admission, date of operation, number of the doctor who conducted the operation, number of the operation theater in which operation was carried out, type of operation, patient's condition before and after operation, treatment advice, etc.

**Constraint:** Patient number should exist in PAT\_ENTRY table. Department, doctor number should exist or should be valid.

**11.ROOM\_DETAILS:** It contains details of all rooms in the hospital. The details stored in this table include room number, room type (general or private), status (whether occupied or not), if occupied, then patient number, patient name, charges per day, etc.

**Constraint:** Room number should be unique. Room type can only be G or P and status can only be Y or N.

### **CaseStudy2:** Railway Reservation

**Aim:** The railway reservations system facilitates the passenger to enquire about the trains available on the basis of source and destination, booking and cancellation of tickets, enquire about the status of the booked ticket, etc. The aim of case study is to design and develop a database maintaining the records of different trains, train status, and passengers. The record of train includes its number, name, source, destination, and days on which it is available, whereas record of train status includes dates for which tickets can be booked, total number of seats available, and number of seats already booked. The database has been developed and tested on the Oracle.

### **Description:**

Passengers can book their tickets for the train in which seats are available. For this, passenger must provide the desired train number and the date for which ticket is to be booked. Before booking a ticket for a passenger, the validity of train number and booking date is checked. Once the train number and booking date are validated, it is checked whether the seat is available. If yes, the ticket is booked with confirm status and corresponding ticket ID is generated which is stored along with other details of the passenger. After all the available tickets are booked, certain numbers of tickets are booked with waiting status. If waiting lot is also finished, then tickets are not booked and a message of non-availability of seats is displayed. The ticket once booked can be cancelled at any time. For this, the passenger must provide the ticket ID (the unique key). The ticket ID is searched, and the corresponding record is deleted. With this, the first ticket with waiting status also gets confirmed.

### **List of Assumption**

Since the reservation system is very large in reality, it is not feasible to develop the case study to that extent and prepare documentation at that level. Therefore, a small sample

case study has been created to demonstrate the working of the reservation system. To implement this sample case study, some assumptions have been made, which are as follows:

1. The number of trains has been restricted to 5.
2. The booking is open only for next seven days from the current date.
3. Only two categories of tickets can be booked, namely, AC and General.
4. The total number of tickets that can be booked in each category AC and General is 10.
5. The total number of tickets that can be given the status of waiting is 2.
6. The in-between stop page stations and their bookings are not considered.

### **Description of Tables and Procedures:**

Tables and procedures that will be created are as follows:

**1. TrainList:** This table consists of details about all the available trains. The information stored in this table includes train number, train name, source, destination, fair for AC ticket, fair for general ticket, and weekdays on which train is available.

**Constraint:** The train number is unique.

**2. Train\_Status:** This table consists of details about the dates on which ticket can be booked for a train and the status of the availability of tickets. The information stored in this table includes train number, train date, total number of AC seats, total number of general seats, number of AC seats booked, and number of general seats booked.

**Constraint:** Train number should exist in Train List table.

**3. Passenger:** This table consists of details about the booked tickets. The information stored in this table includes ticket ID, train number, date for which ticket is booked, name, age, sex and address of the passenger, status of reservation (either confirmed or waiting), and category for which ticket is booked.

**Constraint:** Ticket ID is unique and the train number should exist in Train List table.

**4. Booking:** In this procedure, the train number, train date, and category is read from the passenger. On the basis of the values provided by the passenger, corresponding record is retrieved from the Train\_Status table. If the desired category is AC, then total number of AC seats and number of booked AC seats are compared in order to find whether ticket can be booked or not. Similarly, it can be checked for the general category. If ticket can be booked, then passenger details are read and stored in the Passenger table.

**5. Cancel:** In this procedure, ticket ID is read from the passenger and corresponding record is searched in the Passenger table. If the record exists, it is deleted from the table. After deleting the record (if it is confirmed), first record with waiting status for the same train and same category searched from the Passenger table and its status is changed to confirm.

### **CaseStudy3: Painting Hire Business System Description**

A local business woman has decided to start her own Internet business, called Masterpieces Ltd, Hiring paintings to private individuals and commercial companies. Because of your reputation database designer, she has called upon your services to design and implement a database to support her new business. At the initial planning meeting, to discuss the design, the following

user requirements were requested. The system must be able to manage the details of customers, paintings and those paintings currently on hire to customers. Customers are categorized as B(bronze), S (silver), G (gold) or P (platinum). These categories entitle a customer to a discount of 0%, 5%, 10% or 15% respectively.

Customers often request paintings by a particular artist or theme (e.g. animal, landscape, seascape, naval, still-life, etc). Over time a customer may hire the same painting more than once.

Each painting is allocated a customer monthly rental price defined by the owner. The owner of the painting is then paid 10% of that customer rental price. Any paintings that are not hired within six months are returned to the owner. However, after three months, an owner may re-submit a returned painting. Each painting can only have one artist associated with it. Several reports are required from the system. Three main ones are: For each customer, are reports showing an overview of all the paintings they have hired or are currently hiring.

For each artist, are reports of all paintings submitted for hire or each artist, are reports for those paintings not hired over the past six months to remember to identify key attributes and any foreign key attributes.

**Text Books/Suggested Reading:**

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

L	T	P	Cr.
0	1	2	2

**Prerequisites:** Introduction to Programming

**Course Objectives:** The main objectives of the course are to

1. Introduce core programming concepts of Python programming language.
2. Demonstrate about Python data structures like Lists, Tuples, Sets and dictionaries
3. Implement Functions, Modules and Regular Expressions in Python Programming and to create practical and contemporary applications using these

**Course Outcomes:** After learning the contents of this course, the student must be able to

**CO1:** Implement the core programming concepts of Python programming language. **(Apply-L3)**

**CO2:** Demonstrate about Python data structures like Lists, Tuples, Sets and dictionaries **(Apply-L3)**

**CO3:** Implement Functions, Modules and Regular Expressions in Python Programming and to create practical and contemporary applications. **(Apply-L3)**

**CO4:** Improve individual / teamwork skills, communication & report writing skills with ethical values

### **SYLLABUS:**

#### **UNIT-I:**

History of Python Programming Language, Thrust Areas of Python, Installing Anaconda Python Distribution, Installing and Using Jupyter Notebook.

Parts of Python Programming Language: Identifiers, Keywords, Statements and Expressions, Variables, Operators, Precedence and Associativity, Data Types, Indentation, Comments, Reading Input, Print Output, Type Conversions, the type () Function and Is Operator, Dynamic and Strongly Typed Language.

Control Flow Statements: if statement, if-else statement, if...elif...else, Nested if statement, while Loop, for Loop, continue and break Statements, Catching Exceptions Using try and except Statement.

#### **Sample Experiments:**

1. Write a program to find the largest element among three Numbers.
2. Write a Program to display all prime numbers within an interval
3. Write a program to swap two numbers without using a temporary variable.
4. Demonstrate the following Operators in Python with suitable examples.
  - i) Arithmetic Operators
  - ii) Relational Operators
  - iii) Assignment Operators
  - iv) Logical Operators
  - v) Bit wise Operators
  - vi) Ternary Operator
  - vii) Membership Operators
  - viii) Identity Operators
5. Write a program to add and multiply complex numbers
6. Write a program to print multiplication table of a given number.

**UNIT-II:**

Functions: Built-In Functions, Commonly Used Modules, Function Definition and Calling the function, return Statement and void Function, Scope and Lifetime of Variables, Default Parameters, Keyword Arguments, \*args and \*\*kwargs, Command Line Arguments.

Strings: Creating and Storing Strings, Basic String Operations, Accessing Characters in String by Index Number, String Slicing and Joining, String Methods, Formatting Strings.

Lists: Creating Lists, Basic List Operations, Indexing and Slicing in Lists, Built-In Functions Used on Lists, List Methods, del Statement.

**Sample Experiments:**

1. Write a program to define a function with multiple return values.
2. Write a program to define a function using default arguments.
3. Write a program to find the length of the string without using any library functions.
4. Write a program to check if the substring is present in a given string or not.
5. Write a program to perform the given operations on a list: i. Addition ii. Insertion iii. slicing
6. Write a program to perform any 5 built-in functions by taking any list.

**UNIT-III:**

Dictionaries: Creating Dictionary, Accessing and Modifying key:value Pairs in Dictionaries, Built-In Functions Used on Dictionaries, Dictionary Methods, del Statement.

Tuples and Sets: Creating Tuples, Basic Tuple Operations, tuple() Function, Indexing and Slicing in Tuples, Built-In Functions Used on Tuples, Relation between Tuples and Lists, Relation between Tuples and Dictionaries, Using zip() Function, Sets, Set Methods, Frozenset.

**Sample Experiments:**

1. Write a program to create tuples (name, age, address, college) for at least two members and concatenate the tuples and print the concatenated tuples.
2. Write a program to count the number of vowels in a string (No control flow allowed).
3. Write a program to check if a given key exists in a dictionary or not.
4. Write a program to add a new key-value pair to an existing dictionary.
5. Write a program to sum all the items in a given dictionary.

**UNIT-IV:**

Files: Types of Files, Creating and Reading Text Data, File Methods to Read and Write Data, Reading and Writing Binary Files, Pickle Module, Reading and Writing CSV Files, Python os and os.path Modules.

Object-Oriented Programming: Classes and Objects, Creating Classes in Python, Creating Objects in Python, Constructor Method, Classes with Multiple Objects, Class Attributes Vs Data Attributes, Encapsulation, Inheritance, Polymorphism.

**Sample Experiments:**

1. Write a program to sort words in a file and put them in another file. The output file should have only lower-case words, so any upper-case words from source must be lowered.
2. Python program to print each line of a file in reverse order.
3. Python program to compute the number of characters, words and lines in a file.

4. Write a program to create, display, append, insert and reverse the order of the items in the array.
5. Write a program to add, transpose and multiply two matrices.
6. Write a Python program to create a class that represents a shape. Include methods to calculate its area and perimeter. Implement subclasses for different shapes like circle, triangle, and square.

#### **UNIT-V:**

Introduction to Data Science: Functional Programming, JSON and XML in Python, NumPy with Python, Pandas.

#### **Sample Experiments:**

1. Python program to check whether a JSON string contains complex object or not.
2. Python Program to demonstrate NumPy arrays creation using array () function.
3. Python program to demonstrate use of ndim, shape, size, dtype.
4. Python program to demonstrate basic slicing, integer and Boolean indexing.
5. Python program to find min, max, sum, cumulative sum of array
6. Create a dictionary with at least five keys and each key represent value as a list where this list contains at least ten values and convert this dictionary as a pandas data frame and explore the data through the data frame as follows:
7. Apply head () function to the pandas data frame
8. Perform various data selection operations on Data Frame
9. Select any two columns from the above data frame, and observe the change in one attribute with respect to other attribute with scatter and plot operations in matplotlib

#### **Reference Books:**

1. Gowrishankar S, Veena A., Introduction to Python Programming, CRC Press. Python Programming, S Sridhar, J Indumathi, V M Hariharan, 2ndEdition, Pearson, 2024
2. Introduction to Programming Using Python, Y. Daniel Liang, Pearson.

**Online Learning Resources/Virtual Labs:** <https://www.coursera.org/learn/python-for-applied-data-science-ai> <https://www.coursera.org/learn/python?specialization=python#syllabus>

L	T	P	Cr.
1	0	2	2

**B.Tech. (IV Sem.)      23ME57-DESIGN THINKING & INNOVATION**

**Course Objectives:** The objectives of the course are to

- Bring awareness on innovative design and new product development.
- Explain the basics of design thinking.
- Familiarize the role of reverse engineering in product development.
- Train how to identify the needs of society and convert into demand.
- Introduce product planning and product development process.

**Course Outcomes:**

**CO1:** Define the concepts related to design thinking. **(Remember-L1)**

**CO2:** Explain the fundamentals of Design Thinking and innovation. **(Understand-L2)**

**CO3:** Apply the design thinking techniques for solving problems in various sectors. **(Apply-L3)**

**CO4:** Analyze to work in a multidisciplinary environment. **(Analyze-L4)**

**CO5:** Evaluate the value of creativity. **(Evaluate-L5)**

### UNIT – I

#### Introduction to Design Thinking

Introduction to elements and principles of Design, basics of design-dot, line, shape, form as fundamental design components. Principles of design. Introduction to design thinking, history of Design Thinking, New materials in Industry.

### UNIT - II

#### Design Thinking Process

Design thinking process (empathize, analyze, idea & prototype), implementing the process in driving inventions, design thinking in social innovations. Tools of design thinking - person, costumer, journey map, brainstorming, product development

Activity: Every student presents their idea in three minutes, Every student can present design process in the form of flow diagram or flow chart etc. Every student should explain about product development.

### UNIT - III

#### Innovation

Art of innovation, Difference between innovation and creativity, role of creativity and innovation in organizations. Creativity to Innovation. Teams for innovation, Measuring the impact and value of creativity.

Activity: Debate on innovation and creativity, Flow and planning from idea to innovation, Debate on value-based innovation.

### UNIT - IV

#### Product Design

Problem formation, introduction to product design, Product strategies, Product value, Product planning, product specifications. Innovation towards product design Case studies.

Activity: Importance of modeling, how to set specifications, Explaining their own product design.

### UNIT – V

#### Design Thinking in Business Processes

Design Thinking applied in Business & Strategic Innovation, Design Thinking principles that redefine business – Business challenges: Growth, Predictability, Change, Maintaining Relevance,

Extreme competition, Standardization. Design thinking to meet corporate needs. Design thinking for Startups. Defining and testing Business Models and Business Cases. Developing & testing prototypes.

Activity: How to market our own product, about maintenance, Reliability and plan for startup.

**Textbooks:**

1. Tim Brown, Change by design, 1/e, Harper Bollins, 2009.
2. Idris Mootee, Design Thinking for Strategic Innovation, 1/e, Adams Media, 2014.

**Reference Books:**

1. David Lee, Design Thinking in the Classroom, Ulysses press, 2018.
2. Shrrutin N Shetty, Design the Future, 1/e, Norton Press, 2018.
3. William lidwell, Kritinaholden, &Jill butter, Universal principles of design, 2/e, Rockport Publishers, 2010.
4. Chesbrough.H, The era of open innovation, 2003.

**Online Learning Resources:**

- <https://nptel.ac.in/courses/110/106/110106124/>
- <https://nptel.ac.in/courses/109/104/109104109/>
- [https://swayam.gov.in/nd1\\_noc19\\_mg60/preview](https://swayam.gov.in/nd1_noc19_mg60/preview)
- [https://onlinecourses.nptel.ac.in/noc22\\_de16/preview](https://onlinecourses.nptel.ac.in/noc22_de16/preview)

L	T	P	Cr.
2	0	0	0

B.Tech. (IV Sem.)

23MC01-ENVIRONMENTAL SCIENCE

**Course Objectives:**

- To make the students to get awareness on environment
- To understand the importance of protecting natural resources, ecosystems for future generations and pollution causes due to the day to day activities of human life
- To save earth from the inventions by the engineers.

**Course Outcomes:**

**CO1:** Grasp multidisciplinary nature of environmental studies and various renewable and non-renewable resources. **(Understand-L2)**

**CO2:** Understand flow and bio-geo-chemical cycles and ecological pyramids. **(Understand-L2)**

**CO3:** Understand various causes of pollution and solid waste management and related preventive measures. **(Understand-L2)**

**CO4:** About the rainwater harvesting, watershed management, ozone layer depletion and waste land reclamation. **(Understand-L2)**

**CO5:** Casus of population explosion, value education and welfare programmes. **(Understand- L2)**

**UNIT-I**

Multidisciplinary Nature Of Environmental Studies: – Definition, Scope and Importance – Need for Public Awareness.

Natural Resources: Renewable and non-renewable resources – Natural resources and associated problems – Forest resources – Use and over – exploitation, deforestation, case studies – Timber extraction – Mining, dams and other effects on forest and tribal people – Water resources – Use and over utilization of surface and ground water – Floods, drought, conflicts over water, dams – benefits and problems–Mineral resources: Use and exploitation, environmental effects of extracting and using mineral resources, case studies– Food resources: World food problems, changes caused by agriculture and overgrazing, effects of modern agriculture, fertilizer-pesticide problems, water logging, salinity, case studies.–Energy resources:

**UNIT-II**

Ecosystems: Concep to fan ecosystem.–Structure and function of an ecosystem–Producers, consumers and decomposers – Energy flow in the ecosystem – Ecological succession – Food chains, food webs and ecological pyramids–Introduction, types, characteristic features, structure and function of the following ecosystem:

- Forest ecosystem.
- Grassl and ecosystem
- Desert ecosystem
- Aquatic ecosystems (ponds, streams, lakes, rivers, oceans, estuaries)

Biodiversity And Its Conservation: Introduction Definition: genetic, species and ecosystem diversity–Bio-geographical classification of India–Value of biodiversity: consumptive use, Productive use, social, ethical, aesthetic and option values – Biodiversity at global, National and local levels – India as a mega-diversity nation – Hot-spots of biodiversity – Threats to biodiversity: habitat loss, poaching of wildlife, man-wildlife conflicts– Endangered and endemic species of India –Conservation of biodiversity: In-situ and Ex-situ conservation of biodiversity.

**UNIT-III**

Environmental Pollution: Definition, Cause, effects and control measures of:

- Air Pollution.
- Water pollution
- Soil pollution
- Marine pollution

- e. Noise pollution
- f. Thermal pollution
- g. Nuclear hazards

Solid Waste Management: Causes, effects and control measures of urban and industrial wastes – Role of an individual in prevention of pollution – Pollution case studies – Disaster management: floods, earthquake, cyclone and land slides.

#### UNIT-IV

Social Issues and the Environment: From Unsustainable to Sustainable development– Urban problems related to energy – Water conservation, rain water harvesting, watershed management – Resettlement and rehabilitation of people; its problems and concerns. Case studies – Environmental ethics: Issues and possible solutions–Climate change, global warming, acid rain, ozone layer depletion, nuclear accidents and holocaust. Case Studies – Wastel and reclamation. – Consumerism and waste products. – Environment Protection Act. – Air (Prevention and Control of Pollution) Act. –Water (Prevention and control of Pollution) Act–Wild life Protection Act– Forest Conservation Act–Issues involved in enforcement of environment allegislation–Public awareness.

#### UNIT-V

Human Population And The Environment: Population growth, variation among nations. Population explosion – Family Welfare Programmes. – Environment and human health – Human Rights – Value Education–HIV/AIDS–Women and Child Welfare–Role of information Technology in Environment and human health–Case studies. Field Work: Visit to a local area to document environmental assets River/ forest grassland/ hill/ mountain – Visit to a local polluted site–Urban/Rural/Industrial/Agricultural Study of common plants, insects, and birds–river, hills lopes,etc..

#### Text books:

1. Text book of Environmental Studies for Undergraduate Courses ErachBharucha for University Grants Commission, Universities Press.
2. Palani swamy,“Environmental Studies”,Pearson education
3. S.Azeem Unnisa, “Environmental Studies” Academic Publishing Company
4. K.Raghavan Nambiar,“Text book of Environmental Studies for Undergraduate Courses as per UGC model syllabus”, Scitech Publications (India), Pvt.Ltd.

#### Reference Books:

1. Deeksha Dave and E.Sai Baba Reddy, “Text book of Environmental Science”,Cengage Publications.
2. M.Anji Reddy,“Text book of Environmental Sciences and Technology”, BS Publication.
3. P.Sharma, Comprehensive Environmental studies, Laxmi publications.
4. J. Glynn Henry and Gary W.Heinke, “Environmental Sciences and Engineering”, Prentice Hall of India Private limited
5. G.R.Chatwal, “A Text Book of Environmental Studies ”Himalaya Publishing House
6. Gilbert M.Masters and WendellP.Ela, “Introduction to Environmental Engineering and Science, Prentice Hall of India Private limited.

L	T	P	Cr.
3	0	0	3

B.Tech. (V Sem.)

23IT04-ADVANCED JAVA

**Course Objectives:**

The objectives of this course are to introduce

- The development of dynamic web applications using server-side technologies and Java Database Connectivity (JDBC).
- Java networking concepts and Remote Method Invocation (RMI) using Java APIs.
- The use of modern Java frameworks such as spring for building scalable and maintainable web applications.

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

**CO1:** Understand the architecture and components of JDBC and implement database-driven applications using JDBC. **(Understand - L2)**

**CO2:** Describe J2EE architecture and design structured web applications using J2EE containers and web components. **(Understand - L2)**

**CO3:** Develop web applications using Servlets by managing sessions, filters, and event handling. **(Apply - L3)**

**CO4:** Create dynamic web pages using Java Server Pages (JSP) with scripting elements, JSTL, and expression language. **(Apply - L3)**

**CO5:** Design and build scalable web applications using Spring Framework modules such as Spring MVC, Spring AOP, and Spring DAO. **(Apply - L3)**

**Unit I:**

**JDBC Programming:** JDBC Architecture, Types of JDBC Drivers, Introduction to major JDBC Classes and Interface, creating simple JDBC Application, Types of Statement (Statement Interface, Prepared Statement, Callable Statement), Exploring ResultSet Operations, Batch Updates in JDBC, Creating CRUD Application, Using Rowsets Objects, Managing Database Transaction.

**Unit II:**

**J2EE and Web Development:** J2EE Architecture Types, J2EE Containers, Types of Servers in J2EE Application, HTTP Protocols and API, Request Processing in Web Application, Web Application Structure, Web Containers and Web Architecture Models.

**Unit III:**

**Servlet API and Overview:** Servlet Introduction, Servlet Life Cycle(SLC), Types of Servlet, Servlet Configuration with Deployment Descriptor, Working with ServletContext and ServletConfig Object, Attributes in Servlet, Response and Redirection using Request Dispatcher and using sendRedirect Method, Filter API, Manipulating Responses using Filter API, Session Tracking: using Cookies, HttpSession, Hidden Form Fields and URL Rewriting, Types of Servlet Event: Context Level and Session Level.

**Unit IV:**

**Java Server Pages (JSP):** Introduction to JSP , Comparison with Servlet, JSP Architecture, JSP: Life Cycle, Scripting Elements, Directives, Action Tags, Implicit Objects, Expression Language(EL), JSP Standard Tag Libraries(JSTL), Custom Tag, Session Management, Exception Handling, CRUD Application.

**Unit V:**

**Java Web Frameworks:** Spring MVC Spring: Introduction, Architecture, Spring MVC Module, Life Cycle of Bean Factory, Explore: Constructor Injection, Dependency Injection, Inner Beans, Aliases in Bean, Bean Scopes, Spring Annotations, Spring AOP Module, Spring DAO, Database Transaction Management, CRUD Operation using DAO and Spring API.

**Text Books:**

1. Black Book “Java server programming” J2EE, 1st ed., Dream Tech Publishers, 2008.
2. Complete Reference J2EE, James Keogh, McGraw Hill publication
3. Professional Java Server Programming, Subrahmanyam Allamaraju, Cedric Buest, Wiley Publication
4. Spring in Action, 3rd edition, Craig walls, Manning Publication

**Reference Books:**

1. Core Java, Volume II: Advanced Features, Cay Horstmann, Gary Cornell Pearson Publication
2. 2. JDBC™ API Tutorial and Reference, Third Edition, Maydene Fisher, Jon Ellis, Jonathan Bruce, Addison Wesley
3. 3. Beginning JSP, JSF and Tomcat, Giulio Zambon, Apress

**e-Resources:**

- 1) [https://infyspringboard.onwingspan.com/web/en/viewer/video/lex\\_auth\\_01384313561056051231666\\_shared?collectionType=Collection&collectionId=lex\\_auth\\_01384313888944947231668\\_shared&pathId=](https://infyspringboard.onwingspan.com/web/en/viewer/video/lex_auth_01384313561056051231666_shared?collectionType=Collection&collectionId=lex_auth_01384313888944947231668_shared&pathId=)
1. [https://infyspringboard.onwingspan.com/web/en/viewer/video/lex\\_auth\\_01384330259306086435241\\_shared?collectionType=Course&pathId=lex\\_auth\\_01384329650771558435242\\_shared&collectionId=lex\\_auth\\_01384330035036979236002\\_shared](https://infyspringboard.onwingspan.com/web/en/viewer/video/lex_auth_01384330259306086435241_shared?collectionType=Course&pathId=lex_auth_01384329650771558435242_shared&collectionId=lex_auth_01384330035036979236002_shared)

L	T	P	Cr.
3	0	0	3

B.Tech. (V Sem.)

23CS07-COMPUTER NETWORKS

**Course Objectives:**

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

**Course Outcomes:**

**CO1:** Understand various network types, topologies, reference models, and transmission media **(Understand-L2)**

**CO2:** Examine data link layer design issues, framing techniques, error control, and flow control mechanisms. **(Analyze-L4)**

**CO3:** Apply multiple media access control techniques and evaluate Ethernet standards for network communication. **(Apply-L3)**

**CO4:** Implement routing algorithms, congestion control techniques, and IP addressing schemes for efficient network communication. **(Apply-L3)**

**CO5:** Utilize transport layer protocols (UDP & TCP) and application layer services (HTTP, DNS, Email) to enable secure and reliable data communication. **(Apply-L3)**

**UNIT I: Introduction:** Network Types, LAN, MAN, WAN, Network Topologies Reference models- The OSI Reference Model- the TCP/IP Reference Model - A Comparison of the OSI and TCP/IP Reference Models

**Physical Layer** –Introduction to Guided Media- Twisted-pair cable, Coaxial cable and Fiber optic cable and introduction about unguided media.

**UNIT II: Data link layer:** Design issues, **Framing:** fixed size framing, variable size framing, flow control, error control, error detection and correction codes, CRC, Checksum: idea, one's complement internet checksum, services provided to Network Layer, **Elementary Data Link Layer protocols:** simplex protocol, Simplex stop and wait, Simplex protocol for Noisy Channel. **Sliding window protocol:** One bit, Go back N, Selective repeat-Stop and wait protocol

**UNIT – III: Media Access Control: Random Access:** ALOHA, Carrier sense multiple access (CSMA), CSMA with Collision Detection, CSMA with Collision Avoidance, **Channelization:** frequency division multiple Access(FDMA), time division multiple access(TDMA), code division multiple access(CDMA). **Wired LANs:** Ethernet, Ethernet Protocol, Standard Ethernet, Fast Ethernet(100 Mbps), Gigabit Ethernet, 10 Gigabit Ethernet.

**UNIT – IV: The Network Layer Design Issues** – Store and Forward Packet Switching-Services Provided to the Transport layer- Implementation of Connectionless Service-Implementation of Connection Oriented Service- Comparison of Virtual Circuit and Datagram Networks, Routing Algorithms-The Optimality principle-Shortest path, Flooding, Distance vector, Link state, Hierarchical, Congestion Control algorithms-General principles of congestion control, Congestion prevention polices, Traffic Control Algorithm-Leaky bucket & Token bucket.

**Internet Working:** Network layer in the internet – IP protocols-IP Version 4 protocol-IPV4 Header Format, IP addresses, Class full Addressing, CIDR, Subnets-IP Version 6- The main IPV6 header, Transition from IPV4 to IPV6, Comparison of IPV4 & IPV6.

**UNIT –V: The Transport Layer:** Transport layer protocols: Introduction-services- port number- User data gram protocol-User datagram-UDP services-UDP applications-Transmission control protocol: TCP services- TCP features- Segment- A TCP connection

**Application Layer** – World Wide Web: HTTP, Electronic mail-Architecture- web based mail- email security- TELENET-local versus remote Logging-Domain Name System.

**Text Books:**

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

**References Books:**

1. Data Communications and Networks- Achut S Godbole, AtulKahate Computer Networks, Mayank Dave, CENGAGE

**23CS12-AUTOMATA THEORY & COMPILER  
DESIGN**

L	T	P	Cr.
3	0	0	3

**B.Tech. (V Sem.)**

**Pre-requisites:** Discrete Mathematics, Programming Basics, Data Structures, Algorithms

**Course Objectives:**

1. Understand fundamental concepts of automata, formal languages, and computation.
2. Design finite automata, pushdown automata, and Turing machines.
3. Apply regular expressions and grammars for language processing.
4. Learn the structure and functioning of a compiler.
5. Implement lexical analysis, parsing, and intermediate code generation.

**Course Outcomes:**

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

**CO1:** Understand and apply concepts of formal languages, alphabets, strings, and finite automata for modelling simple computational problems.

**CO2:** Construct and analyze regular expressions and context-free grammars for language definition and pattern matching.

**CO3:** Design pushdown automata and Turing machines and evaluate problems of decidability and language classification.

**CO4:** Explain the phases of compilation and implement lexical and syntax analysis techniques using tools and algorithms.

**CO5:** Apply syntax-directed translation and generate intermediate code with memory management for a target machine.

**UNIT – I: Introduction to Finite Automata:** Structural Representations, Automata and Complexity, the Central Concepts of Automata Theory – Alphabets, Strings, Languages, Problems. Nondeterministic Finite Automata: Formal Definition, an application, Text Search, Finite Automata with Epsilon-Transitions. **Deterministic Finite Automata:** Definition of DFA, How A DFA Process Strings, The language of DFA, Conversion of NFA with  $\epsilon$ -transitions to NFA without  $\epsilon$ -transitions. Conversion of NFA to DFA

**UNIT – II: Regular Expressions:** Finite Automata and Regular Expressions, Applications of Regular Expressions, Algebraic Laws for Regular Expressions, Conversion of Finite Automata to Regular Expressions. Pumping Lemma for Regular Languages: Statement of the pumping lemma, Applications of the Pumping Lemma. **Context-Free Grammars:** Definition of Context-Free Grammars, Derivations Using a Grammar, Leftmost and Rightmost Derivations, the Language of a Grammar, Parse Trees, Ambiguity in Grammars and Languages.

**UNIT – III: Push Down Automata:** Definition of the Pushdown Automaton, the Languages of a PDA, Equivalence of PDA's and CFG's, Acceptance by final state Turing Machines: Introduction to Turing Machine, Formal Description, Instantaneous description, The language of a Turing machine Undecidability: Undecidability, A Language that is Not Recursively Enumerable, An Undecidable Problem That is RE, Undecidable Problems about Turing Machines

**UNIT - IV Introduction:** The structure of a compiler, Lexical Analysis: The Role of the Lexical Analyzer, Input Buffering, Recognition of Tokens, The Lexical- Analyzer Generator Lex, Syntax Analysis: Introduction, Context-Free Grammars, Writing a Grammar, Top-Down Parsing, Bottom- Up Parsing, Introduction to LR Parsing: Simple LR, More Powerful LR Parsers R18 B.Tech. CS&D Syllabus JNTU Hyderabad

**UNIT - V Syntax-Directed Translation:** Syntax-Directed Definitions, Evaluation Orders for SDD's, Syntax Directed Translation Schemes, Implementing L-Attributed SDD's. Intermediate-Code Generation: Variants of Syntax Trees, Three-Address Code Run-Time Environments: Stack Allocation of Space, Access to Nonlocal Data on the Stack, Heap Management.

**TEXT BOOKS:**

1. Introduction to Automata Theory, Languages, and Computation, 3rd Edition, John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman, Pearson Education.
2. Compilers: Principles, Techniques and Tools, Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman, 2nd Edition, Pearson.
3. Theory of Computer Science – Automata languages and computation, Mishra and Chandra shekaran, 2nd Edition, PHI.

**REFERENCE BOOKS:**

1. Introduction to Formal languages Automata Theory and Computation, Kamala Krithivasan, Rama R, Pearson.
2. Introduction to Languages and The Theory of Computation, John C Martin, TMH.
3. lex & yacc – John R. Levine, Tony Mason, Doug Brown, O'reilly
4. Compiler Construction, Kenneth C. Loudon, Thomson. Course Technology.

B.Tech. (V Sem.)

**23IT03-OBJECT ORIENTED ANALYSIS AND DESIGN (PROFESSIONAL ELECTIVE-I)**

L	T	P	Cr.
3	0	0	3

**Course Objectives:** The main objective is the students to

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

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

**CO1.** Understand the structure and attributes of complex systems. (Understand - L2)

**CO2.** Understand the Unified Modeling Language (UML) and its significance in object-oriented software development. (Understand - L2)

**CO3.** Apply object-oriented modeling techniques to design class and object diagrams, including advanced structural concepts like interfaces, types, and packages. (Apply - L3)

**CO4.** Develop and interpret basic behavioral models such as use cases, interaction diagrams, and activity diagrams for real-world applications. Apply - L3)

**CO5.** Design advanced behavioral and architectural models using state machines and component diagrams to represent complex software systems. (Apply - L3)

**UNIT-I:**

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

**UNIT-II:**

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

**UNIT-III:**

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

**UNIT-IV:**

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

**UNIT-V:**

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

**Text Books:**

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

**Reference Books:**

1. Meilir Page-Jones: Fundamentals of Object-Oriented Design in UML, Pearson Education.
2. Pascal Roques: Modeling Software Systems Using UML2, WILEY- Dreamtech India Pvt. Ltd.
3. Atul Kahate: Object Oriented Analysis & Design, The McGraw-Hill Companies.
4. Applying UML and Patterns: An introduction to Object-Oriented Analysis and Design and Unified Process, Craig Larman, Pearson Education.

B.Tech. (V Sem.)

**23IT05-CYBER SECURITY  
(PROFESSIONAL ELECTIVE-I)**

L	T	P	Cr.
3	0	0	3

**Course Objectives:**

The aim of the course is to

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

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

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

**UNIT**

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

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

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

**Text Books:**

1. Sunit Belapure Nina Godbole “Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives”, WILEY, 2011.
2. Nelson Phillips and Enfinger Stuart, “Computer Forensics and Investigations”, Cengage Learning, New Delhi, 2009.

**Reference Books:**

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

**E-Resources:**

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

B.Tech. (V Sem.)

**23AD02-ARTIFICIAL INTELLIGENCE  
(PROFESSIONAL ELECTIVE-I)**

L	T	P	Cr.
3	0	0	3

**Pre-requisite:**

1. Knowledge in Computer Programming.
2. A course on “Mathematical Foundations of Computer Science”.
3. Background in linear algebra, data structures and algorithms, and probability.

**Course Objectives:**

1. The student should be made to study the concepts of Artificial Intelligence.
2. The student should be made to learn the methods of solving problems using Artificial Intelligence.
3. The student should be made to introduce the concepts of Expert Systems.
4. To understand the applications of AI, namely game playing, theorem proving, and machine learning.
5. To learn different knowledge representation techniques

**UNIT - I**

**Introduction:** AI problems, foundation of AI and history of AI intelligent agents: Agents and Environments, the concept of rationality, the nature of environments, structure of agents, problem solving agents, problem formulation.

**UNIT - II**

**Searching-** Searching for solutions, uniformed search strategies – Breadth first search, depth first Search. Search with partial information (Heuristic search) Hill climbing, A\* ,AO\* Algorithms, Problem reduction, Game Playing-Adversial search, Games, mini-max algorithm, optimal decisions in multiplayer games, Problem in Game playing, Alpha-Beta pruning, Evaluation functions.

**UNIT - III**

**Representation of Knowledge:** Knowledge representation issues, predicate logic- logic programming, semantic nets- frames and inheritance, constraint propagation, representing knowledge using rules, rules based deduction systems. Reasoning under uncertainty, review of probability, Bayes’ probabilistic interferences and Dempster Shafer theory.

**UNIT - IV**

**Logic concepts:** First order logic. Inference in first order logic, propositional vs. first order inference, unification & lifts forward chaining, Backward chaining, Resolution, Learning from observation Inductive learning, Decision trees, Explanation based learning, Statistical Learning methods, Reinforcement Learning.

**UNIT - V**

**Expert Systems:** Architecture of expert systems, Roles of expert systems – Knowledge Acquisition Meta knowledge Heuristics. Typical expert systems – MYCIN, DART, XCON: Expert systems shells.

**Textbooks:**

1. S. Russel and P. Norvig, “Artificial Intelligence – A Modern Approach”, Second Edition, Pearson Education.
2. Kevin Night and Elaine Rich, Nair B., “Artificial Intelligence (SIE)”, Mc Graw Hill

**Reference Books:**

1. David Poole, Alan Mackworth, Randy Goebel, “Computational Intelligence: a logical approach”, Oxford University Press.
2. G. Luger, “Artificial Intelligence: Structures and Strategies for complex problemsolving”, Fourth Edition, Pearson Education.
3. J. Nilsson, “Artificial Intelligence: A new Synthesis”, Elsevier Publishers.
4. Artificial Intelligence, SarojKaushik, CENGAGE Learning.

**E-Resources:**

1. <https://ai.google/>
2. [https://swayam.gov.in/nd1\\_noc19\\_me71/preview](https://swayam.gov.in/nd1_noc19_me71/preview)

**23AD04-DATA WAREHOUSING & DATA MINING****B.Tech. (V Sem.)****(PROFESSIONAL ELECTIVE-I)**

L	T	P	Cr.
3	0	0	3

**Pre-requisites:** Data Structures, Algorithms, Probability & Statistics, Data Base Management Systems

**Course Objectives:** The main objective of the course is to

- Introduce basic concepts and techniques of data warehousing and datamining
- Examine the types of data to be mined and apply pre-processing methods on raw data
- Discover interesting patterns, analyze supervised and unsupervised models and estimate the accuracy of the algorithms.

**Course Outcomes**

**CO1:** Design data warehouses to support effective data modeling, integration, and analytical processing.

**CO2:** Understand data preprocessing techniques required to convert raw data into a suitable format for effective machine learning applications.

**CO3:** Apply classification techniques using different algorithms to solve real-world problems and evaluate their performance.

**CO4:** Apply Apriori and FP-Growth algorithms to analyze frequent patterns and uncover insights from large datasets.

**CO5:** Understand clustering concepts and various cluster analysis methods to group similar data points effectively.

**UNIT-I:** Data Warehousing and Online Analytical Processing: Basic concepts, Data Warehouse Modeling: Data Cube and OLAP, Data Warehouse Design and Usage, Data Warehouse Implementation, Cloud Data Warehouse, Data Mining and Pattern Mining, Technologies, Applications, Major issues, Data Objects & Attribute Types, Basic Statistical Descriptions of Data, Data Visualization, Measuring Data Similarity and Dissimilarity. (**Textbook- 1**)

**UNIT II:** Data Preprocessing: An Overview, Data Cleaning, Data Integration, Data Reduction, Data Transformation and Data Discretization. (**Textbook- 1**)

**UNIT-III:** Classification: Basic Concepts, General Approach to solving a classification problem, Decision Tree Induction: Attribute Selection Measures, Tree Pruning, Scalability and Decision Tree Induction, Visual Mining for Decision Tree Induction, Bayesian Classification Methods: Bayes Theorem, Naïve Bayes Classification, Rule-Based Classification, Model Evaluation and Selection. (**Textbook- 2**)

**UNIT-IV:** Association Analysis: Problem Definition, Frequent Itemset Generation, Rule Generation: Confident Based Pruning, Rule Generation in Apriori Algorithm, Compact Representation of frequent item sets, FP-Growth Algorithm. (**Textbook- 2**)

**UNIT-V:** Cluster Analysis: Overview, Basics and Importance of Cluster Analysis, Clustering techniques, Different Types of Clusters; K-means: The Basic K-means Algorithm, K-means Additional Issues, Bi-secting K Means, Agglomerative Hierarchical Clustering: Basic Agglomerative Hierarchical Clustering Algorithm DBSCAN: Traditional Density Center-Based Approach, DBSCAN Algorithm, Strengths and Weaknesses. (**Textbook- 2**)

**Textbooks:**

1. Data Mining concepts and Techniques, 3rd edition, Jiawei Han, Michel Kamber, Elsevier, 2011.
2. Introduction to Data Mining: Pang-Ning Tan & Michael Steinbach, Vipin Kumar, Pearson, 2012.

**Reference Books:**

1. Data Mining: VikramPudi and P. Radha Krishna, Oxford Publisher.
2. Data Mining Techniques, Arun K Pujari, 3rd edition, UniversitiesPress,2013.
3. (NPTEL course by Prof.PabitraMitra)[http://onlinecourses.nptel.ac.in/noc17\\_mg24/preview](http://onlinecourses.nptel.ac.in/noc17_mg24/preview)
4. [http://www.saedsayad.com/data\\_mining\\_map.htm](http://www.saedsayad.com/data_mining_map.htm)

L	T	P	Cr.
0	0	3	1.5

**B.Tech. (V Sem.)****23IT54-ADVANCED JAVA LAB****Course Objectives:**

The objectives of this lab course are to introduce

- To make use of Servlet and JSP API in the process of enterprise application deployment.
- Implement components such as JSTL
- Distinguish Application Server, Web Container, JDBC
- Design and Development of web application having collaboration of Servlets, JSPs, Spring.

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

**CO-1:** Implement database operations using JDBC with different types of statements and manage transactions effectively. **(Apply - L3)**

**CO-2:** Develop and deploy server-side programs using Servlets for handling client requests and session tracking. **(Apply - L3)**

**CO-3:** Create dynamic and interactive web pages using JSP and JSTL for data display and user interaction. **(Apply - L3)**

**CO-4:** Demonstrate the use of various JSTL tags including core, format, function, and SQL tags in JSP applications. **(Apply - L3)**

**CO-5:** Design and implement MVC-based web applications using the Spring Framework with database and transaction support. **(Apply - L3)**

1. Write a JDBC application which will interact with Database and perform the following task.
  - a. Create Student Table with RollNo, Name, and Address field and insert few records.
  - b. Using Statement Object display the content of Record.
  - c. Using Statement Object Insert Two Record.
  - d. Using Statement Object Update One Record.
  - e. Using Statement Object Delete One Record.
  - f. Using Statement Object display the content of Record.
2. Write a JDBC application which will interact with Database and perform the following task.
  - a. Create Student Table with RollNo, Name, and Address field and insert few records.
  - b. Using PreparedStatement Object display the content of Record.
  - c. Using PreparedStatement Object Insert Two Record.
  - d. Using PreparedStatement Object Update One Record.
  - e. Using PreparedStatement Object Delete One Record.
  - f. Using PreparedStatement Object display the content of Record
3. Write a JDBC application which will interact with Database and perform the following task.
  - a. Create a store procedure which will insert one record into employee table.
  - b. Create a store procedure which will retrieve salary for given employee id.
  - c. Write a java application which will call the above procedure and display appropriate information on screen
4. Design a JDBC application which will demonstrate Scrollable ResultSet functionality.
5. Design a JDBC application which will demonstrate Updatable ResultSet functionality.
6. Write down the Program for testing the Servlet and study deployment descriptor.
7. Write down the program for testing the include action for servlet collaboration.
8. Create login form and perform state management using Cookies, HttpSession and URL Rewriting.
9. Write down the Program which displays the simple JSP file
10. Write down the program in which input the two numbers in an html file and then display the addition in JSP file.

11. Perform Database Access through JSP.
12. Write down a program which demonstrates the core tag of JSTL.
13. Write down a program which demonstrates the Format tag of JSTL.
14. Write down a program which demonstrates the Function tag of JSTL.
15. Write down a program which demonstrates the SQL tag of JSTL.
16. Study and Implement MVC using Spring Framework
17. Using Spring Template manage Database and Transaction.

**E-Resources:**

- 2) [https://infyspringboard.onwingspan.com/web/en/viewer/video/lex\\_auth\\_01384313561056051231666\\_shared?collectionType=Collection&collectionId=lex\\_auth\\_01384313888944947231668\\_shared&pathId=](https://infyspringboard.onwingspan.com/web/en/viewer/video/lex_auth_01384313561056051231666_shared?collectionType=Collection&collectionId=lex_auth_01384313888944947231668_shared&pathId=)
- 3) [https://infyspringboard.onwingspan.com/web/en/viewer/video/lex\\_auth\\_01384330259306086435241\\_shared?collectionType=Course&pathId=lex\\_auth\\_01384329650771558435242\\_shared&collectionId=lex\\_auth\\_01384330035036979236002\\_shared](https://infyspringboard.onwingspan.com/web/en/viewer/video/lex_auth_01384330259306086435241_shared?collectionType=Course&pathId=lex_auth_01384329650771558435242_shared&collectionId=lex_auth_01384330035036979236002_shared)

L	T	P	Cr.
0	0	3	1.5

**Course Objectives:**

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

**Course Outcomes:** This Course will enable students to

**CO1:** Apply fundamental networking concepts by configuring network devices, analyzing protocols, and implementing data link layer techniques. **(Apply – L3)**

**CO2:** Implement various network algorithms, including error detection, congestion control, routing, and shortest path computation. **(Apply – L3)**

**CO3:** Analyze network traffic, security vulnerabilities, and performance metrics using tools like Wireshark, Nmap, and NS2 Simulator. **(Apply – L3)**

**CO4:** Improve individual / teamwork skills, communication & report writing skills with ethical values.

**List of Experiments:**

1. Study of Network devices in detail and connect the computers in Local Area Network.
2. Write a Program to implement the data link layer framing methods such as
  - i) Character stuffing ii) bit stuffing.
3. Write a Program to implement data link layer framing method checksum.
4. Write a program for Hamming Code generation for error detection and correction.
5. Write a Program to implement on a data set of characters the three CRC polynomials – CRC-12, CRC-16
6. Write a Program to implement Sliding window protocol for Goback N.
7. Write a Program to implement Sliding window protocol for Selective repeat.
8. Write a Program to implement Stop and Wait Protocol.
9. Write a program for congestion control using leaky bucket algorithm
10. Write a Program to implement Distance vector routing algorithm by obtaining routing table at each node (Take an example subnet graph with weights indicating delay between nodes).
11. Wire shark
  - i) Packet Capture Using Wire shark
  - ii) Starting Wire shark
  - iii) Viewing Captured Traffic
  - iv) Analysis and Statistics & Filters.
12. Do the following using NS3 Simulator
  - i) NS3 Simulator-Introduction
  - ii) Simulate to Find the Number of Packets Dropped
  - iii) Simulate to Find the Number of Packets Dropped by TCP/UDP
  - iv) Simulate to Find the Number of Packets Dropped due to Congestion
  - v) Simulate to Compare Data Rate& Throughput.

**B.Tech. (V Sem.)**                      **23ITS1-PYTHON WITH DJANGO**  
**(SKILL ENHANCEMENT COURSE)**

L	T	P	Cr.
0	1	2	2

**Course Objectives:**

The main objectives of the course are to

- Design and build static as well as dynamic web pages and interactive web-based applications
- Web development using Django framework.
- Analyze and create functional website in Django and deploy Django Web Application on Cloud

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

**CO1:** Apply Python libraries and web frameworks to design and develop dynamic and interactive web applications. **.(Apply - L3)**

**CO2:** Develop and integrate Django-based modules to implement user authentication, database operations, and UI components. **.(Apply - L3)**

**CO3:** Deploy functional Django web applications to cloud platforms by applying standard deployment practices and tools. **.(Apply - L3)**

**CO4:** Improve individual / teamwork skills, communication & report writing skills with ethical values. **(Apply-L3)**

**UNIT-I : Python libraries for web development :**

Collections-Container datatypes, Tkinter-GUI applications, Requests-HTTP requests, BeautifulSoup4-web scraping, Scrapy, Zappa, Dash, CherryPy, Turbo Gears, Flask, Web2Py, Bottle, Falcon, Cubic Web, Quixote, Pyramid.

**Sample Experiments:**

1. Write a Python GUI program to import Tkinter package and create a window. Set its title and add a label to the window.
2. Write a Python program that designs a simple login form with labels and Entry widgets, arranging them in a grid using the Grid geometry manager.
3. Write a program using BeautifulSoup4 library for web scraping for a given URL 4.Develop a sample Hello World page using Flask framework 5.Develop a sample web page using CherryPy / Web2Py / Bottle Framework

**UNIT-II: Introduction to Django Framework**

Understanding Django environment, Features of Django and Django architecture, MVC and MTV, Urls and Views, Mapping the views to URLs, Django Template, Template inheritance Django Models, Creating model for site, Converting the model into a table, Fields in Models, Integrating Bootstrap into Django, Creating tables, Creating grids, Creating carousels.

**Sample Experiments:**

1. Create a Sample “Hello World” Application using Django
2. Create a Login and Registration Page using MVC architecture in Django Framework
3. Create a sample page in Django by integrating Bootstrap.
4. Create an application with Tables, grids in Django 5.Create a Django App with Carousels feature.

**UNIT-III : Integrating Accounts & Authentication on Django**

Introduction to Django Authentication System, Security Problem &Solution with Django Creating Registration Form using Django, Adding Email Field in Forms, Configuring email settings, Sending emails with Django, Adding Grid Layout On Registration Page, Adding Page Restrictions, Login Functionality Test and Logout.

**Sample Experiments:**

1. Create a registration page using Authentication System
2. Create an application in Django to send emails using email settings and Grid Layout
3. Create an application in Django using page restriction / authentication with Login and Logout Functionality
4. Create a sample form using Django Forms

**UNIT-IV: Connecting SQLite with Django**

DatabaseMigrations,FetchDataFromDatabase,DisplayingDataOnTemplates,AddingCondition On Data, Sending data from url to view, Sending data from view to template, Saving objects into database, Sorting objects, Filtering objects, Deleting objects, Difference between session and cookie, Creating sessions and cookies in Django

**Sample Experiments:**

1. Create an app in Django which fetches data from database and show as list and also save objects in database
2. Create an app in Django for performing CRUD operations on records in a database
3. Create an app in Django which uses session management and cookies to store and manage user sessions.

**UNIT-V: Deploying Django Web Application on Cloud**

Creating a functional website in Django, Four Important Pillars to Deploy, registering on Heroku and GitHub, Push project from Local System to GitHub, working with Django Heroku, Working with StaticRoot, Handling WSGI with gunicorn, setting up Database & adding users.

**Sample Experiments:**

1. Create a website in Django with login, and registration page.
2. Register on GitHub, and Heroku and deploy the website on Heroku with all the functionalities developed.
3. Configure Django to handle static files.

**Text books:**

1. Martin Brown, "Python: The Complete Reference Paper back", 4th Edition 2018, McGraw Hill Education.
2. Reema Thareja, "Python Programming: Using Problem Solving Approach", 3rd Edition 2017, Oxford.
3. Daniel Rubio, Apress, "Beginning Django Web Application Development and Deployment with Python", 2nd Edition 2017,Apress.

**Reference Books:**

1. Tom Aratyn, "Building Django 2.0 Web Applications: Create enterprise-grade, scalable Python web applications easily with Django 2.0",2ndEdition 2018, Packt Pub
2. Harry Percival, "Test-Driven Development with Python: Obey the Testing Goat: Using Django, Selenium and JavaScript",2nd Edition 2019, Kindle Edition.

**23IT53-USER INTERFACE DESIGN USING  
FLUTTER**

L	T	P	Cr.
0	0	2	1

**B.Tech. (V Sem.)**

**Course Objectives:**

- Learns to Implement Flutter Widgets and Layouts
- Understands Responsive UI Design and with Navigation in Flutter
- Knowledge on Widges and customize widgets for specific UI elements, Themes
- Understand to include animation apart from fetching data

**List of Experiments:**

Students need to implement the following experiments

1. a) Install Flutter and Dart SDK.  
b) Write a simple Dart program to understand the language basics.
2. a) Explore various Flutter widgets (Text, Image, Container, etc.).  
b) Implement different layout structures using Row, Column, and Stack widgets.
3. a) Design a responsive UI that adapts to different screen sizes.  
b) Implement media queries and breakpoints for responsiveness.
4. a) Set up navigation between different screens using Navigator.  
b) Implement navigation with named routes.
5. a) Learn about stateful and stateless widgets.  
b) Implement state management using set State and Provider.
6. a) Create custom widgets for specific UI elements.  
b) Apply styling using themes and custom styles.
7. a) Design a form with various input fields.  
b) Implement form validation and error handling.
8. a) Add animations to UI elements using Flutter's animation framework.  
b) Experiment with different types of animations (fade, slide, etc.).
9. a) Fetch data from a REST API.  
b) Display the fetched data in a meaningful way in the UI.
10. a) Write unit tests for UI components.  
b) Use Flutter's debugging tools to identify and fix issues.

**Text Books:**

1. Marco L. Napoli, Beginning Flutter: A Hands-on Guide to App Development.
2. Rap Payne, Beginning App Development with Flutter: Create Cross-Platform Mobile Apps 1<sup>st</sup> Edition, Apres
3. Richard Rose, Flutter & Dart Cookbook, Developing Full stack Applications for the Cloud, Oreilly.

L	T	P	Cr.
3	0	0	3

**Course Objectives:**

- To explain the evolving utility computing model called cloud computing.
- To introduce the various levels of services offered by cloud.
- To discuss the fundamentals of cloud enabling technologies such as distributed computing, service-oriented architecture and virtualization.
- To emphasize the security and other challenges in cloud computing.
- To introduce the advanced concepts such as containers, serverless computing and cloud-centric Internet of Things.

**UNIT -I:** Introduction to Cloud Computing Fundamentals

Cloud computing at a glance, defining a cloud, cloud computing reference model, types of services (IaaS, PaaS, SaaS), cloud deployment models (public, private, hybrid), utility computing, cloud computing characteristics and benefits, cloud service providers (Amazon Web Services, Microsoft Azure, Google AppEngine).

**UNIT-II:** Cloud Enabling Technologies

Ubiquitous Internet, parallel and distributed computing, elements of parallel computing, hardware architectures for parallel computing (SISD, SIMD, MISD, MIMD), elements of distributed computing, Inter-process communication, technologies for distributed computing, remote procedure calls (RPC), service-oriented architecture (SOA), Web services, virtualization.

**UNIT-III:** Virtualization and Containers

Characteristics of virtualized environments, taxonomy of virtualization techniques, virtualization and cloud Computing, pros and cons of virtualization, technology examples (XEN, VMware), building blocks of containers, container platforms (LXC, Docker), container orchestration, Docker Swarm and Kubernetes, public cloud VM (e.g. Amazon EC2) and container (e.g. Amazon Elastic Container Service) offerings.

**UNIT-IV:** Cloud computing challenges

Economics of the cloud, cloud interoperability and standards, scalability and fault tolerance, energy efficiency in clouds, federated clouds, cloud computing security, fundamentals of computer security, cloud security architecture, cloud shared responsibility model, security in cloud deployment models.

**UNIT -V:** Advanced concepts in cloud computing

Serverless computing, Function-as-a-Service, serverless computing architecture, public cloud (e.g. AWS Lambda) and open-source (e.g. OpenFaaS) serverless platforms, Internet of Things (IoT), applications, cloud-centric IoT and layers, edge and fog computing, DevOps, infrastructure-as-code, quantum cloud computing.

**Text Books:**

1. Mastering Cloud Computing, 2<sup>nd</sup> edition, Rajkumar Buyya, Christian Vecchiola, ThamaraiSelvi, ShivanandaPoojara, Satish N. Srirama, Mc Graw Hill, 2024.
2. Distributed and Cloud Computing, Kai Hwang, Geoffrey C. Fox, Jack J. Dongarra, Elsevier, 2012.

**Reference Books:**

1. Cloud Computing, Theory and Practice, Dan C Marinescu, 2<sup>nd</sup> edition, MK Elsevier, 2018.
2. Essentials of cloud Computing, K. Chandrasekhran, CRC press, 2014.
3. Online documentation and tutorials from cloud service providers (e.g., AWS, Azure, GCP)

**23CS13-CRYPTOGRAPHY & NETWORK  
SECURITY**

L	T	P	Cr.
3	0	0	3

**B.Tech. (VI Sem.)**

**Course Objectives:**

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

**UNIT I:**

**Basic Principles: Security Goals, Cryptographic Attacks, Services and Mechanisms, Mathematics of Cryptography-** integer arithmetic, modular arithmetic, matrices, linear congruence.

**UNIT II:**

**Symmetric Encryption:** Mathematics of Symmetric Key Cryptography-algebraic structures,  $GF(2^n)$  Fields, Introduction to Modern Symmetric Key Ciphers-modern block ciphers, modern stream ciphers, Data Encryption Standard- DES structure, DES analysis, Security of DES, Multiple DES, Advanced Encryption Standard-transformations, key expansions, AES ciphers, Analysis of AES.

**UNIT III:**

**Asymmetric Encryption:** Mathematics of Asymmetric Key Cryptography-primes, primality testing, factorization, CRT, Asymmetric Key Cryptography- RSA crypto system, Rabin cryptosystem, Elgamal Crypto system, ECC

**UNIT IV:**

**Data Integrity, Digital Signature Schemes & Key Management :**Message Integrity and Message Authentication-message integrity, Random Oracle model, Message authentication, Cryptographic Hash Functions-whirlpool, SHA-512, Digital Signature- process, services, attacks, schemes, applications, Key Management-symmetric key distribution, Kerberos.

**UNIT V:**

**Network Security-I:** Security at application layer: PGP and S/MIME, Security at the Transport Layer: SSL and TLS, **Network Security-II :**Security at the Network Layer: IPsec-two modes, two security protocols, security association, IKE, ISAKMP, System Security-users, trust, trusted systems, buffer overflow, malicious software, worms, viruses, IDS, Firewalls.

**Text Books:**

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

**Reference Books:**

1. Network Security and Cryptography, 1<sup>st</sup> Edition, Bernard Meneges, Cengage Learning,2018

B.Tech. (VI Sem.)

23AM01-MACHINE LEARNING

L	T	P	Cr.
3	0	0	3

**Course Objectives:**

The objectives of the course is to

- Define machine learning and its different types (supervised and unsupervised) and understand their applications.
- Apply supervised learning algorithms including decision trees and k-nearest neighbours (k-NN).
- Implement unsupervised learning techniques, such as K-means clustering.

**UNIT-I: Introduction to Machine Learning:** Evolution of Machine Learning, Paradigms for ML, Learning by Rote, Learning by Induction, Reinforcement Learning, Types of Data, Matching, Stages in Machine Learning, Data Acquisition, Feature Engineering, Data Representation, Model Selection, Model Learning, Model Evaluation, Model Prediction, Search and Learning, Data Sets.

**UNIT-II: Nearest Neighbor-Based Models:** Introduction to Proximity Measures, Distance Measures, Non-Metric Similarity Functions, Proximity Between Binary Patterns, Different Classification Algorithms Based on the Distance Measures ,K-Nearest Neighbor Classifier, Radius Distance Nearest Neighbor Algorithm, KNN Regression, Performance of Classifiers, Performance of Regression Algorithms.

**UNIT-III: Models Based on Decision Trees:** Decision Trees for Classification, Impurity Measures, Properties, Regression Based on Decision Trees, Bias–Variance Trade-off, Random Forests for Classification and Regression. The Bayes Classifier: Introduction to the Bayes Classifier, Bayes’ Rule and Inference, The Bayes Classifier and its Optimality, Multi-Class Classification, Class Conditional Independence and Naive Bayes Classifier (NBC)

**UNIT-IV: Linear Discriminants for Machine Learning:** Introduction to Linear Discriminants, Linear Discriminants for Classification, Perceptron Classifier, Perceptron Learning Algorithm, Support Vector Machines, Linearly Non-Separable Case, Non-linear SVM, Kernel Trick, Logistic Regression, Linear Regression, Multi-Layer Perceptrons (MLPs), Backpropagation for Training an MLP.

**UNIT-V: Clustering :** Introduction to Clustering, Partitioning of Data, Matrix Factorization, Clustering of Patterns, Divisive Clustering, Agglomerative Clustering, Partitional Clustering, K-Means Clustering, Soft Partitioning, Soft Clustering, Fuzzy C-Means Clustering, Rough Clustering, Rough K-Means Clustering Algorithm, Expectation Maximization-Based Clustering, Spectral Clustering.

**Text Books:**

1. “Machine Learning Theory and Practice”, M N Murthy, V S Ananthanarayana, Universities Press (India), 2024

**Reference Books:**

1. “Machine Learning”, Tom M. Mitchell, McGraw-Hill Publication, 2017
2. “Machine Learning in Action”, Peter Harrington, DreamTech
3. “Introduction to Data Mining”, Pang-Ning Tan, Michel Stenbach, Vipin Kumar, 7<sup>th</sup> Edition, 2019.

**23IT06-SOFTWARE TESTING METHODOLOGIES**  
**(PROFESSIONAL ELECTIVE-II)**

L	T	P	Cr.
3	0	0	3

**B.Tech. (VI Sem.)**

**Course Objectives**

- To provide knowledge of the concepts in software testing such as testing process, criteria, strategies, and methodologies.
- To develop skills in software test automation and management using the latest tools.

**UNIT - I**

Introduction: Purpose of testing, Dichotomies, model for testing, consequences of bugs, taxonomy of bugs  
 Flow graphs and Path testing: Basics concepts of path testing, predicates, path predicates and achievable paths, path sensitizing, path instrumentation, application of path testing.

**UNIT - II**

Transaction Flow Testing: transaction flows, transaction flow testing techniques.

Data Flow testing: Basics of data flow testing, strategies in data flow testing, application of data flow testing.

Domain Testing: domains and paths, Nice & ugly domains, domain testing, domains and interfaces testing, domain and interface testing, domains and testability.

**UNIT - III**

Paths, Path products and Regular expressions: path products & path expression, reduction procedure, applications, regular expressions & flow anomaly detection.

Logic Based Testing: overview, decision tables, path expressions, kv charts, specifications.

**UNIT - IV**

State, State Graphs and Transition testing: state graphs, good & bad state graphs, state testing, Testability tips.

**UNIT - V**

Graph Matrices and Application: Motivational overview, matrix of graph, relations, power of a matrix, node reduction algorithm, building tools. (Student should be given an exposure to a tool like Jmeter/selenium/soapUI/Catalon).

**Text Books:**

1. Software Testing techniques - BarisBeizer, Dreamtech, second edition.
2. Software Testing Tools – Dr. K. V. K. K. Prasad, Dreamtech.

**Reference Books:**

1. The craft of software testing - Brian Marick, Pearson Education.
2. Software Testing Techniques – SPD(Oreille)
3. Software Testing in the Real World – Edward Kit, Pearson.
4. Effective methods of Software Testing, Perry, John Wiley.
5. Art of Software Testing – Meyers, John Wiley.

**B.Tech. (VI Sem.)**                      **23IT09-AUGMENTED REALITY & VIRTUAL REALITY (PROFESSIONAL ELECTIVE-II)**

L	T	P	Cr.
3	0	0	3

**Course Objectives:**

- Provide a foundation to the fast-growing field of AR and make the students aware of the various AR concepts.
- To give historical and modern overviews and perspectives on virtual reality. It describes the fundamentals of sensation, perception, technical and engineering aspects of virtual reality systems.

**UNIT - I**

**Introduction to Augmented Reality:** Augmented Reality - Defining augmented reality, history of augmented reality, Examples, Related fields

**Displays:** Multimodal Displays, Visual Perception, Requirements and Characteristics, Spatial Display Model, Visual Displays

**Tracking:** Tracking, Calibration, and Registration, Coordinate Systems, Characteristics of Tracking Technology, Stationary Tracking Systems, Mobile Sensors

**UNIT - II**

**Computer Vision for Augmented Reality:** Marker Tracking, Multiple-Camera Infrared Tracking, Natural Feature Tracking by Detection, Outdoor Tracking.

**Interaction:** Output Modalities, Input Modalities, Tangible Interfaces, Virtual User Interfaces on Real Surfaces, Augmented Paper, Multi-view Interfaces, Haptic Interaction

**Software Architectures:** AR Application Requirements, Software Engineering Requirements, Distributed Object Systems, Dataflow, Scene Graphs

**UNIT - III**

**Introduction to Virtual Reality:** Defining Virtual Reality, History of VR, Human Physiology and Perception

**The Geometry of Virtual Worlds:** Geometric Models, Axis-Angle Representations of Rotation, Viewing Transformations

**Light and Optics:** Basic Behavior of Light, Lenses, Optical Aberrations, The Human Eye, Cameras, Displays

**UNIT - IV**

**The Physiology of Human Vision:** From the Cornea to Photoreceptors, From Photoreceptors to the Visual Cortex, Eye Movements, Implications for VR

**Visual Perception:** Visual Perception - Perception of Depth, Perception of Motion, **Perception of Color**

**Visual Rendering:** Visual Rendering -Ray Tracing and Shading Models, Rasterization, Correcting Optical Distortions, Improving Latency and Frame Rates, Immersive Photos and Videos

**UNIT - V**

**Motion in Real and Virtual Worlds:** Velocities and Accelerations, The Vestibular System, Physics in the Virtual World, Mismatched Motion and Vection

**Interaction:** Motor Programs and Remapping, Locomotion, Social Interaction

**Audio:** The Physics of Sound, The Physiology of Human Hearing, Auditory Perception, Auditory Rendering

**Text Books:**

1. Augmented Reality: Principles & Practice by Schmalstieg / Hollerer, Pearson Education India; First edition (12 October 2016), ISBN-10: 9332578494
2. Virtual Reality, Steven M. LaValle, Cambridge University Press, 2016

**Reference Books:**

1. Allan Fowler-AR Game Developmentll, 1st Edition, A press Publications, 2018, ISBN 978-1484236178
2. Understanding Virtual Reality: Interface, Application and Design, William R Sherman and Alan B Craig, (The Morgan Kaufmann Series in Computer Graphics)". Morgan Kaufmann Publishers, San Francisco, CA, 2002
3. Developing Virtual Reality Applications: Foundations of Effective Design, Alan B Craig, William R Sherman and Jeffrey D Will, Morgan Kaufmann, 2009
4. Designing for Mixed Reality, Kharis O'Connell Published by O'Reilly Media, Inc., 2016, ISBN:9781491962381
5. SanniSiltanen- Theory and applications of marker-based augmented reality. Julkaisija – Utgivare Publisher. 2012. ISBN 978-951-38-7449-0
6. Gerard Jounghyun Kim, "Designing Virtual Systems: The Structured Approach", 2005

## 23IT08-DEVOPS

B.Tech. (VI Sem.)

(PROFESSIONAL ELECTIVE-II)

L	T	P	Cr.
3	0	0	3

**Course Objectives:**

The main objectives of this course are to:

- Describe the agile relationship between development and IT operations.
- Understand the skill sets and high-functioning teams involved in DevOps and related methods to reach a continuous delivery capability.
- Implement automated system update and DevOps lifecycle.

**UNIT-I**

**Introduction to DevOps:** Introduction to SDLC, Agile Model. Introduction to Devops. DevOps Features, DevOps Architecture, DevOps Lifecycle, Understanding Workflow and principles, Introduction to DevOps tools, Build Automation, Delivery Automation, Understanding Code Quality, Automation of CI/CD. Release management, Scrum, Kanban, delivery pipeline, bottlenecks, examples

**UNIT-II**

**Source Code Management (GIT):**The need for source code control, The history of source code management, Roles and code, source code management system and migrations. What is Version Control and GIT, GIT Installation, GIT features, GIT workflow, working with remote repository, GIT commands, GIT branching, GIT staging and collaboration. **UNIT TESTING - CODE COVERAGE:** Junit, NUnit & Code Coverage with Sonar Qube, SonarQube - Code Quality Analysis.

**UNIT-III**

**Build Automation - Continuous Integration (CI):** Build Automation, What is CI Why CI is Required, CI tools, Introduction to Jenkins (With Architecture), jenkins workflow, jenkins master slave architecture, Jenkins Pipelines, PIPELINE BASICS - Jenkins Master, Node, Agent, and Executor Freestyle Projects & Pipelines, Jenkins for Continuous Integration, Create and Manage Builds, User Management in Jenkins Schedule Builds, Launch Builds on Slave Nodes.

**UNIT-IV**

**Continuous Delivery (CD):** Importance of Continuous Delivery, CONTINUOUS DEPLOYMENT CD Flow, Containerization with Docker: Introduction to Docker, Docker installation, Docker commands, Images & Containers, DockerFile, Running containers, Working with containers and publish to Docker Hub.

**Testing Tools:** Introduction to Selenium and its features, JavaScript testing.

**UNIT-V**

**Configuration Management - ANSIBLE:** Introduction to Ansible, Ansible tasks, Roles, Jinja templating, Vaults, Deployments using Ansible.

**CONTAINERIZATION USING KUBERNETES(OPENSHIFT):** Introduction to Kubernetes Namespace & Resources, CI/CD - On OCP, BC, DC & ConfigMaps, Deploying Apps on Openshift Container Pods. Introduction to Puppet master and Chef.

**Text Books:**

1. Joyner, Joseph., Devops for Beginners: Devops Software Development Method Guide for Software Developers and It Professionals, 1<sup>st</sup> Edition MihailsKonoplows, 2015.
2. Alisson Machado de Menezes., Hands-on DevOps with Linux, 1<sup>st</sup> Edition, BPB Publications, India, 2021.

**Reference Books:**

1. Len Bass, Ingo Weber, Liming Zhu. DevOps: A Software Architect's Perspective. Addison Wesley; ISBN-10
2. Gene Kim Je Humble, Patrick Debois, John Willis. The DevOps Handbook, 1st Edition, IT Revolution Press, 2016.
3. Verona, Joakim Practical DevOps, 1<sup>st</sup> Edition, Packt Publishing, 2016.
4. Joakim Verona. Practical Devops, Ingram short title; 2<sup>nd</sup> edition (2018). ISBN10: 1788392574
5. Deepak Gaikwad, Viral Thakkar. DevOps Tools from Practitioner's Viewpoint. Wiley publications. ISBN: 9788126579952

B.Tech. (VI Sem.)

**23IT10-GENERATIVE AI  
(PROFESSIONAL ELECTIVE-II)**

L	T	P	Cr.
3	0	0	3

**Course Objectives:**

- Understand the basics of Generative AI.
- Know the basics of Text Generation.
- Understand the process of generating videos.
- Know about GAN and its variants.

**UNIT I :**

Introduction To Gen Ai: Historical Overview of Generative modelling, Difference between Gen AI and Discriminative Modeling, Importance of generative models in AI and Machine Learning, Types of Generative models, GANs, VAEs, autoregressive models and Vector quantized Diffusion models, Understanding if probabilistic modeling and generative process, Challenges of Generative Modeling, Future of Gen AI, Ethical Aspects of AI, Responsible AI, Use Cases.

**UNIT II:**

Generative Models For Text: Language Models Basics, Building blocks of Language models, Transformer Architecture, Encoder and Decoder, Attention mechanisms, Generation of Text, Models like BERT and GPT models, Generation of Text, Autoencoding, Regression Models, Exploring ChatGPT, Prompt Engineering: Designing Prompts, Revising Prompts using Reinforcement Learning from Human Feedback (RLHF), Retrieval Augmented Generation, Multimodal LLM, Issues of LLM like hallucination.

**UNIT III:**

Generation of Images: Introduction to Generative Adversarial Networks, Adversarial Training Process, Nash Equilibrium, Variational Autoencoders, Encoder-Decoder Architectures, Stable Diffusion Models, Introduction to Transformer-based Image Generation, CLIP, Visual Transformers ViT- Dall-E2 and Dall-E3, GPT-4V, Issues of Image Generation models like Mode Collapse and Stability.

**UNIT IV:**

Generation of Painting, Music, and Play: Variants of GAN, Types of GAN, Cyclic GAN, Using Cyclic GAN to Generate Paintings, Neural Style Transfer, Style Transfer, Music Generating RNN, MuseGAN, Autonomous agents, Deep Q Algorithm, Actor-critic Network.

**UNIT V:**

Open Source Models And Programming Frameworks: Training and Fine tuning of Generative models, GPT 4 All, Transfer learning and Pretrained models, Training vision models, Google Copilot, Programming LLM, LangChain, Open Source Models, Llama, Programming for TimeSformer, Deployment, Hugging Face.

**Text Books:**

1. Denis Rothman, "Transformers for Natural Language Processing and Computer Vision", Third Edition, Packt Books, 2024

**Reference Books:**

1. David Foster, "Generative Deep Learning", O'Reily Books, 2024.
2. Altaf Rehmani, "Generative AI for Everyone", BlueRose One, 2024.

**23IT07-SOFTWARE PROJECT MANAGEMENT**  
**(PROFESSIONAL ELECTIVE-III)**

L	T	P	Cr.
3	0	0	3

**B.Tech. (VI Sem.)**

**Course Objectives:**

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

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

**UNIT-I:**

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

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

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

**The old way and the new:** The principles of conventional software Engineering, principles of modern software management, transitioning to an iterative process.

**UNIT-II:**

**Life cycle phases:** Engineering and production stages, inception, Elaboration, construction, transition phases.

**Artifacts of the process:** The artifact sets, Management artifacts, Engineering artifacts, programmatic artifacts.

**UNIT- III:**

**Model based software architectures:** A Management perspective and technical perspective.

**Work Flows of the process:** Software process workflows, Iteration workflows.

**Checkpoints of the process:** Major mile stones, Minor Milestones, Periodic status assessments.

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

**UNIT- IV:**

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

**Process Automation:** Automation Building blocks, The Project Environment.

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

**UNIT-V:**

Agile Methodology, ADAPting to Scrum, Patterns for Adopting Scrum, Iterating towards Agility.

**Fundamentals of DevOps:** Architecture, Deployments, Orchestration, Need, Instance of applications, DevOps delivery pipeline, DevOps eco system. DevOps adoption in projects: Technology aspects, Agiling capabilities, Tool stack implementation, People aspect, processes

**Text Books:**

1. Software Project Management, Walker Royce, PEA, 2005.
2. Succeeding with Agile: Software Development Using Scrum, Mike Cohn, Addison Wesley.
3. The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations, Gene Kim , John Willis , Patrick Debois , Jez Humb,1st Edition, O'Reilly publications, 2016.

**Reference Books:**

1. Software Project Management, Bob Hughes,3/e, Mike Cotterell, TMH
2. Software Project Management, Joel Henry, PEA
3. Software Project Management in practice, Pankaj Jalote, PEA, 2005,
4. Effective Software Project Management, Robert K.Wysocki, Wiley,2006.
5. Project Management in IT, Kathy Schwalbe, Cengage

B.Tech. (VI Sem.)

**23CS14-MOBILE ADHOC NETWORKS  
(PROFESSIONAL ELECTIVE-III)**

L	T	P	Cr.
3	0	0	3

**Course Objectives:**

From the course the student will learn

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

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

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

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

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

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

**Text Books:**

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

**Reference Books:**

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

**23AM05-NATURAL LANGUAGE PROCESSING**  
**(PROFESSIONAL ELECTIVE-III)**

L	T	P	Cr.
3	0	0	3

**B.Tech. (VI Sem.)**

**Course Objectives:**

This course introduces the fundamental concepts and techniques of natural language processing (NLP).

- Students will gain an in-depth understanding of the computational properties of natural languages and the commonly used algorithms for processing linguistic information.
- The course examines NLP models and algorithms using both the traditional symbolic and the more recent statistical approaches.
- Enable students to be capable to describe the application based on natural language processing and to show the points of syntactic, semantic and pragmatic processing.

**UNIT I:**

**INTRODUCTION:** Origins and challenges of NLP – Language Modeling: Grammar-based LM, Statistical LM – Regular Expressions, Finite-State Automata – English Morphology, Transducers for lexicon and rules, Tokenization, Detecting and Correcting Spelling Errors, Minimum Edit Distance.

**UNIT II:**

**WORD LEVEL ANALYSIS:** Unsmoothed N-grams, Evaluating N-grams, Smoothing, Interpolation and Backoff – Word Classes, Part- of-Speech Tagging, Rule-based, Stochastic and Transformation-based tagging, Issues in PoS tagging – Hidden Markov and Maximum Entropy models.

**UNIT III:**

**SYNTACTIC ANALYSIS:** Context-Free Grammars, Grammar rules for English, Treebanks, Normal Forms for grammar – Dependency Grammar – Syntactic Parsing, Ambiguity, Dynamic Programming parsing – Shallow parsing Probabilistic CFG, Probabilistic CYK, Probabilistic Lexicalized CFGs – Feature structures, Unification of feature structures

**UNIT IV:**

**SEMANTICS AND PRAGMATICS:** Requirements for representation, First-Order Logic, Description Logics – Syntax-Driven Semantic analysis, Semantic attachments – Word Senses, Relations between Senses, Thematic Roles, selectional restrictions – Word Sense Disambiguation, WSD using Supervised, Dictionary & Thesaurus, Bootstrapping methods – Word Similarity using Thesaurus and Distributional methods.

**UNIT V:**

**DISCOURSE ANALYSIS AND LEXICAL RESOURCES:** Discourse segmentation, Coherence – Reference Phenomena, Anaphora Resolution using Hobbs and Centering Algorithm – Coreference Resolution – Resources: Porter Stemmer, Lemmatizer, Penn Treebank, Brill’s Tagger, WordNet, PropBank, FrameNet, Brown Corpus, British National Corpus (BNC).

**Text Books:**

1. Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech, 2<sup>nd</sup> Edition, Daniel Jurafsky, James H. Martin -Pearson Publication, 2014.
2. Natural Language Processing with Python, First Edition, Steven Bird, Ewan Klein and Edward Loper, OReilly Media, 2009.

**Reference Books:**

1. Language Processing with Java and Ling Pipe Cookbook, 1<sup>st</sup> Edition, Breck Baldwin, Atlantic Publisher, 2015.
2. Natural Language Processing with Java, 2<sup>nd</sup> Edition, Richard M Reese, OReilly Media, 2015.
3. Handbook of Natural Language Processing, Second, Nitin Indurkha and Fred J. Damerau, Chapman and Hall/CRC Press, 2010. Edition
4. Natural Language Processing and Information Retrieval, 3<sup>rd</sup> Edition, Tanveer Siddiqui, U.S. Tiwary, Oxford University Press, 2008.

**23CS09-DISTRIBUTED OPERATING SYSTEM  
(PROFESSIONAL ELECTIVE-III)**

L	T	P	Cr.
3	0	0	3

**B.Tech. (VI Sem.)**

**Course Objectives:**

The main objective of the course is to introduce design issues and different message passing techniques in DOS, distributed systems, RPC implementation and its performance in DOS, distributed shared memory and resource management, distributed file systems and evaluate the performance in terms of fault tolerance, file replication as major factors

**Unit I:**

**Fundamentals:**

What is Distributed Computing Systems? Evolution of Distributed Computing System; Distributed Computing System Models; what is Distributed Operating System? Issues in Designing a Distributed Operating System; Introduction to Distributed Computing Environment (DCE).

**Message Passing:**

Introduction, Desirable features of a Good Message Passing System, Issues in PC by Message Passing, Synchronization, Buffering, Multi-datagram Messages, Encoding and Decoding of Message Data, Process Addressing, Failure Handling, Group Communication, Case Study: 4.3 BSD UNIX IPC Mechanism.

**Unit II: Remote Procedure Calls:**

Introduction, The RPC Model, Transparency of RPC, Implementing RPC Mechanism, Stub Generation, RPC Messages, Marshaling Arguments and Results, Server Management, Parameter-Passing Semantics, Call Semantics, Communication Protocols for RPCs, Complicated RPCs, Client-Server Binding, Exception Handling, Security, Some Special Types of RPCs, RPC in Heterogeneous Environments, Lightweight RPC, Optimization for Better Performance, Case Studies: Sun RPC

**Unit III: Distributed Shared Memory:**

Introduction, General Architecture of DSM systems, Design and Implementation Issues of DSM, Granularity, Structure of Shared Memory Space, Consistency Models, Replacement Strategy, Thrashing, Other approaches to DSM, Heterogeneous DSM, Advantages of DSM. Synchronization: Introduction, Clock Synchronization, Event Ordering, Mutual Exclusion, Dead Lock, Election Algorithms

**Unit IV: Resource Management:**

Introduction, Desirable Features of a Good Global Scheduling Algorithm, Task Assignment Approach, Load – Balancing Approach, Load – Sharing Approach Process Management: Introduction, Process Migration, Threads.

**Unit V: Distributed File Systems:**

Introduction, Desirable Features of a Good Distributed File System, File models, File–Accessing Models, File – Sharing Semantics, File – Caching Schemes, File Replication, Fault Tolerance, Atomic Transactions and Design Principles.

**Text books**

1. Pradeep. K. Sinha: Distributed Operating Systems: Concepts and Design, PHI, 2007.

**Reference Books:**

1. Andrew S. Tanenbaum: Distributed Operating Systems, Pearson Education, 2013.
2. Ajay D. Kshemkalyani and MukeshSinghal, Distributed Computing: Principles, Algorithms and Systems, Cambridge University Press, 20083. SunitaMahajan, Seema Shan, “ Distributed Computing”, Oxford University Press,2015

B.Tech. (VI Sem.)

23CS59-CLOUD COMPUTING LAB

L	T	P	Cr.
0	0	3	1.5

**Course Objectives:**

- To introduce the various levels of services offered by cloud.
- To give practical knowledge about working with virtualization and containers.
- To introduce the advanced concepts such as serverless computing and cloud simulation.

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

**CO-1:** Demonstrate various service types, delivery models and technologies of a cloud computing environment.

**CO-2:** Distinguish the services based on virtual machines and containers in the cloud offerings.

**CO-3:** Assess the challenges associated with a cloud-based application.

**CO-4:** Discuss advanced cloud concepts such as serverless computing and cloud simulation.

**CO-5:** Examine various programming paradigms suitable to solve real world and scientific problems using cloud services.

**List of Experiments:**

1. Lab on web services
  2. Lab on IPC, messaging, publish/subscribe
  3. Install VirtualBox/VMware Workstation with different flavours of Linux or windows OS on top of windows8 or above.
  4. Install a C compiler in the virtual machine created using VirtualBox and execute Simple Programs.
  5. Create an Amazon EC2 instance and set up a web-server on the instance and associate an IP address with the instance. In the process, create a security group allowing access to port 80 on the instance.
- OR
6. Do the same with OpenStack
  7. Install Google App Engine. Create a hello world app and other simple web applications using python/java.
  8. Start a Docker container and set up a web-server (e.g. apache2 or Python based Flask micro web framework) on the instance. Map the host directory as a data volume for the container.
  9. Find a procedure to transfer the files from one virtual machine to another virtual machine. Similarly, from one container to another container.
  10. Find a procedure to launch virtual machine using trystack (Online Openstack Demo Version)
  11. Install Hadoop single node cluster and run simple applications like word count.
  12. Utilize OpenFaaS – Serverless computing framework and demonstrate basic event driven function invocation.
  13. Simulate a cloud scenario using CloudSim and run a scheduling algorithm that is not present in CloudSim.

**Text Books:**

1. Mastering Cloud Computing, 2<sup>nd</sup> edition, Rajkumar Buyya, Christian Vecchiola, ThamaraiSelvi, Shivananda Poojara, Satish N. Srirama, McGraw Hill, 2024.
2. Distributed and Cloud Computing, Kai Hwang, Geoffrey C. Fox, Jack J. Dongarra, Elsevier, 2012.

**Reference Books:**

1. Cloud Computing, Theory and Practice, Dan C Marinescu, 2<sup>nd</sup> edition, MK Elsevier, 2018.
2. Cloud Computing: Principles and Paradigms by Rajkumar Buyya, James Broberg and Andrzej M. Goscinski, Wiley, 2011.
3. Online documentation and tutorials from cloud service providers (e.g. AWS, Google App Engine)
4. Docker, Reference documentation, <https://docs.docker.com/reference/>
5. Open FaaS, Serverless Functions Made Simple, <https://docs.openfaas.com/>

B.Tech. (VI Sem.)

23AM51-MACHINE LEARNING LAB

L	T	P	Cr.
0	0	3	1.5

**Course Objectives:**

- To learn about computing central tendency measures and Data preprocessing techniques
- To learn about classification and regression algorithms
- To apply different clustering algorithms for a problem.

**Software Required: Python/R/Weka**

Lab should cover the concepts studied in the course work, sample list of Experiments:

1. Compute Central Tendency Measures: Mean, Median, Mode Measure of Dispersion: Variance, Standard Deviation.
2. Apply the following Pre-processing techniques for a given dataset.
  - a. Attribute selection
  - b. Handling Missing Values
  - c. Discretization
  - d. Elimination of Outliers
3. Apply KNN algorithm for classification and regression
4. Demonstrate decision tree algorithm for a classification problem and perform parameter tuning for better results
5. Demonstrate decision tree algorithm for a regression problem
6. Apply Random Forest algorithm for classification and regression
7. Demonstrate Naïve Bayes Classification algorithm.
8. Apply Support Vector algorithm for classification
9. Demonstrate simple linear regression algorithm for a regression problem
10. Apply Logistic regression algorithm for a classification problem
11. Demonstrate Multi-layer Perceptron algorithm for a classification problem
12. Implement the K-means algorithm and apply it to the data you selected. Evaluate performance by measuring the sum of the Euclidean distance of each example from its class center. Test the performance of the algorithm as a function of the parameters K.
13. Demonstrate the use of Fuzzy C-Means Clustering
14. Demonstrate the use of Expectation Maximization based clustering algorithm

**23HSS1-SOFT SKILLS**  
**(SKILL ENHANCEMENT COURSE)**

L	T	P	Cr.
0	1	2	2

**B.Tech. (VI Sem.)**

**Course Objectives:**

- To equip the students with the skills to effectively communicate in English
- To train the students in interview skills, group discussions and presentation skills
- To motivate the students to develop confidence
- To enhance the students' interpersonal skills
- To improve the students' writing skills

**UNIT – I**

**Analytical Thinking & Listening Skills:** Self-Introduction, Shaping Young Minds - A Talk by Azim Premji (Listening Activity), Self – Analysis, Developing Positive Attitude, Perception.

**Communication Skills:** Verbal Communication; Non Verbal Communication (Body Language)

**UNIT – II**

**Self-Management Skills:** Anger Management, Stress Management, Time Management, Six Thinking Hats, Team Building, Leadership Qualities

**Etiquette:** Social Etiquette, Business Etiquette, Telephone Etiquette, Dining Etiquette

**UNIT – III**

**Standard Operation Methods :**Basic Grammars, Tenses, Prepositions, Pronunciation, Letter Writing; Note Making, Note Taking, Minutes Preparation, Email & Letter Writing

**UNIT-IV**

**Job-Oriented Skills:** Group Discussion, Mock Group Discussions, Resume Preparation, Interview Skills, Mock Interviews

**UNIT-V**

**Interpersonal relationships:** Introduction, Importance, Types, Uses, Factors affecting interpersonal relationships, Accommodating different styles, Consequences of interpersonal relationships

**Text books:**

1. Barun K. Mitra, Personality Development and Soft Skills, Oxford University Press, 2011.
2. S.P. Dhanavel, English and Soft Skills, Orient Blackswan, 2010.

**Reference books:**

1. R.S.Aggarwal, A Modern Approach to Verbal & Non-Verbal Reasoning, S.Chand& Company Ltd., 2018.
2. Raman, Meenakshi& Sharma, Sangeeta, Technical Communication Principles and Practice, Oxford University Press, 2011.

**E-resources:**

1. [https://swayam-plus.swayam2.ac.in/courses/course-details?id=P\\_CAMBR\\_01](https://swayam-plus.swayam2.ac.in/courses/course-details?id=P_CAMBR_01)

L	T	P	Cr.
2	0	0	-

**Course Objective:** The course will explain the basic related to writing the technical reports and understanding the concepts related to formatting and structuring the report. This will help students to comprehend the concept of proofreading, proposals and practice

#### Unit I:

**Introduction:** An introduction to writing technical reports, technical sentences formation, using transitions to join sentences, Using tenses for technical writing.

**Planning and Structuring:** Planning the report, identifying reader(s), Voice, Formatting and structuring the report, Sections of a technical report, Minutes of meeting writing.

#### Unit II:

**Drafting report and design issues:** The use of drafts, Illustrations and graphics.

**Final edits:** Grammar, spelling, readability and writing in plain English: Writing in plain English, Jargon and final layout issues, Spelling, punctuation and Grammar, Padding, Paragraphs, Ambiguity.

#### Unit III:

**Proofreading and summaries:** Proofreading, summaries, Activities on summaries. **Presenting final reports:** Printed presentation, Verbal presentation skills, Introduction to proposals and practice.

#### Unit IV: Using word processor:

Adding a Table of Contents, Updating the Table of Contents, Deleting the Table of Contents, Adding an Index, Creating an Outline, Adding Comments, Tracking Changes, Viewing Changes, Additions, and Comments, Accepting and Rejecting Changes, Working with Footnotes and Endnotes, Inserting citations and Bibliography, Comparing Documents, Combining Documents, Mark documents final and make them read only., Password protect Microsoft Word documents., Using Macros,

#### Unit V:

**Nature of Intellectual Property:** Patents, Designs, Trade and Copyright. Process of **Patenting and Development:** technological research, innovation, patenting, development. International Scenario: International cooperation on Intellectual Property

#### Text Books:

1. Kompal Bansal & Parshit Bansal, "Fundamentals of IPR for Beginner's", 1<sup>st</sup> Ed., BS Publications, 2016.
2. William S. Pfeiffer and Kaye A. Adkins, "Technical Communication: A Practical Approach", Pearson.
3. Ramappa, T., "Intellectual Property Rights Under WTO", 2<sup>nd</sup> Ed., S Chand, 2015.

#### Reference Books:

1. Adrian Wallwork, English for Writing Research Papers, Springer New York Dordrecht Heidelberg London, 2011.
2. Day R, How to Write and Publish a Scientific Paper, Cambridge University Press(2006)

#### E-resources:

1. <https://www.udemy.com/course/reportwriting/>
2. <https://www.udemy.com/course/professional-business-english-and-technical-report-writing/>
3. <https://www.udemy.com/course/betterbusinesswriting/>