COURSE STRUCTURE
&
DETAILED SYLLABUS
(R23 Regulation)
For
Bachelor of Technology
I B.Tech. (IT)
(Applicable for Batches Admitted from 2023-2024)

Department of
INFORMATION TECHNOLOGY
(Applicable for Batches Admitted from 2023-2024)

DEPARTMENT OF INFORMATION TECHNOLOGY

COLLEGE OF ENGINEERING, VIZIANAGARAM

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY GURAJADA
VIZIANAGARAM

ANDHRA PRADESH-535003, INDIA

B. Tech (R23) JNTUGV-CEV (Autonomous) w.e.f 2023-24

DEPARTMENT OF INFORMATION TECHNOLOGY

JNTU-GV COLLEGE OF ENGINEERING, VIZIANAGARAM (Autonomous)
JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY:GURAJADA VIZIANAGARAM
VIZIANAGARAM-535003, ANDHRA PRADESH, INDIA.

B.Tech (R23) COURSE STRUCTURE - DEPARTMENT OF INFORMATION TECHNOLOGY

B.Tech - COURSE STRUCTURE – R23
(Applicable from the academic year 2023-24 onwards)

I-B.Tech I-Semester							
S. No.	**Category**	**Course Code**	**Course Name**	**L**	**T**	**P**	**Credits**
1.	BS	R23BS01	Linear Algebra & Calculus	3	0	0	3
2.	BS	R23BS03T	Engineering Physics	3	0	0	3
3.	HS	R23HS01T	Communicative English	2	0	0	2
4.	ES	R23ES01	Basic Civil & Mechanical Engineering	3	0	0	3
5.	ES	R23ES07T	Introduction to Programming	3	0	0	3
6.	HS	R23HS01P	Communicative English Lab	0	0	2	1
7.	BS	R23BS03P	Engineering Physics Lab	0	0	2	1
8.	BS	R23ES02	Engineering Workshop	0	0	3	1.5
9.	ES	R23ES06	IT Workshop	0	0	2	1
10.	ES	R23ES07P	Computer Programming Lab	0	0	3	1.5
11.	Audit	R23MC01	Health and Wellness, Yoga, and Sports	0	0	1	0.5
			Total				**20.5**

Category	**Credits**
Basic Science Course	10
Engineering Science Courses	10
Mandatory Course	0.5
Total Credits	20.5

I-B.Tech II-Semester							
S. No.	**Category**	**Course Code**	**Course Name**	**L**	**T**	**P**	**Credits**
1.	BS	R23BS02	Differential Equations and Vector calculus	3	0	0	3
2.	BS	R23BS05T	Chemistry	3	0	0	3
3.	ES	R23ES03	Engineering Graphics	1	0	4	3
4.	ES	R23ES04	Basic Electrical &Electronics Engineering	3	0	0	3
5.	PC	R23PC04T	Data Structures	3	0	0	3
6.	BS	R23BS05P	Chemistry Lab	0	0	2	1
7.	ES	R23ES05	Electrical &Electronics Engineering workshop	0	0	3	1.5
8.	PC	R23PC04P	Data Structures Lab	0	0	3	1.5
9.	Audit	R23MC02	NSS/NCC/Scouts &Guides/Community Service	0	0	1	0.5
			Total				**19.5**

Category	**Credits**
Basic Science Course	7
Engineering Science Courses	7.5
Professional core Courses	4.5
Mandatory Course	0.5
TOTAL	19.5

IYear-ISemester

L	T	P	C
3	0	0	3

LINEAR ALGEBRA& CALCULUS
(Common for all branches)

Course Objectives:
To equip the students with standard concepts and tools of mathematics to handle various real-world problems and their applications.

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

- Develop matrix algebra techniques that is needed by engineers for practical applications.
- Familiarize with functions of several variables which is useful in optimization.
- Learn important tools of calculus in higher dimensions.
- Familiarize with double and triple integrals of functions of several variables in two and three dimensions.

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, Gauss Seidel Iteration Method.

UNITII Linear Transformation 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.

UNITIII 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.

UNITIV Partial differentiation and Applications (Multi variable calculus)
Partial derivatives, total derivatives, chain rule, change of variables, Taylor's and Maclaurin's series expansion of functions of two variables. Jacobians, maxima and minima of functions of two variables, method of Lagrange multipliers.

UNITV 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. B.S.Grewal,HigherEngineeringMathematics,44/e,KhannaPublishers,2017.
2. Erwin Kreyszig, Advanced Engineering Mathematics, 10/e, John Wiley & Sons, 2018.

Reference Books:

1. R.K.JainandS.R.K.Iyengar,AdvancedEngineeringMathematics,5/e,AlphaScienceInternationalLtd.,2021 (9th reprint).
2. GeorgeB.Thomas,MauriceD.WeirandJoelHass,ThomasCalculus,14/e,PearsonPublishers,2018.
3. GlynJames, Advanced Modern Engineering Mathematics, 5/e, Pearson publishers, 2018.

4. Michael Greenberg, Advanced EngineeringMathematics,9th edition, Pearson edn
5. H. K Das, Er. Rajnish Verma, Higher Engineering Mathematics, S. Chand,2021

L	T	P	C
3	0	0	3

IYear-Isemester

Engineering Physics

COURSE OBJECTIVES

1 Bridging the gap between the Physics in school at 10+2 level and UG level engineeringcourses.

2 To identify the importance of the optical phenomenonie. interference, diffraction and polarization related to its Engineering applications

3 EnlightentheperiodicarrangementofatomsinCrystallinesolidsbyBragg'slaw

4 To explain the significant concepts of dielectric and magnetic materials that leads to potential applications in the emerging micro devices.

5 Enlightenment of the concepts of Quantum Mechanics and to provide fundamentals of deBroglie matter waves, quantum mechanical wave equation and its application, the importance of free electron theory for metals.

6. To Understand the Physics of Semiconductors and their working mechanism,Concept utilization of transport phenomenon of charge carriers in semiconductors.

COURSE OUTCOMES

CO1 **Explain** the need of coherent sources and the conditions for sustained interference (L2). **Identify** the applications of interference in engineering (L3). **Analyze** the differences between interference and diffraction with applications (L4). **Illustrate** the concept of polarization of light and its applications (L2). **Classify** ordinary refracted light and extraordinary refracted rays by their states of polarization (L2)

CO2 **Classify** various crystal systems (L2). **Identify** different planes in the crystal structure (L3). **Analyze** the crystalline structure by Bragg's X-ray diffractometer (L4).

CO3 **Explain** the concept of dielectric constant and polarization in dielectric materials (L2). **Summarize** various types of polarization of dielectrics (L2). **Interpret** Lorentz field and Claussius-Mosotti relation in dielectrics (L2). **Classify** the magnetic materials based on susceptibility and their temperature dependence (L2).

CO4 **Describe** the dual nature of matter (L1). **Explain** the significance of wave function (L2). **Identify** the role of Schrodinger's time independent wave equation in studying particle in one-dimensional infinite potential well (L3). **Identify** the role of classical and quantum freeelectron theory in the study of electrical conductivity (L3).

CO5 **Classify** the crystalline solids (L2).**Outline** the properties of charge carriers in semiconductors (L2). **Identify** the type of semiconductor using Hall effect (L2). **Apply** the concept of effective mass of electron (L3).

Unit-I: Wave Optics **12hrs**

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

Diffraction: Introduction - Fresnel and Fraunhofer diffractions - Fraunhofer diffraction due to single slit, double slit & Diffraction Grating (Qualitative).

Polarization: Introduction -Types of polarization - Polarization by reflection, and Double refraction - Nicol's Prism -Half wave and Quarter wave plates.

Unit Outcomes:

The students will be able to

- **Explain** the need of coherent sources and the conditions for sustained interference (L2)
- **Identify** engineering applications of interference (L3)
- **Illustrate** the concept of polarization of light and its applications (L2)
- **Classify** ordinary polarized light and extraordinary polarized light (L2)

Unit II: Crystallography **8hrs**

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. Bragg's law - X-ray Diffractometer.

Unit Outcomes:

The students will be able to

- **Classify** various crystal systems (L2)
- **Identify** different planes in the crystal structure (L3)
- **Analyze** the crystalline structure by Bragg's X-ray diffractometer (L4)

Unit-III: Dielectric and Magnetic Materials **8hrs**

Dielectric Materials: Introduction - Dielectric polarization - Dielectric polarizability, Susceptibility, Dielectric constant and Displacement Vector - Types of polarizations- Electronic (Quantitative), Ionic (Quantitative) and Orientation polarizations (Qualitative) - Lorentz internal field - Clausius-Mossotti equation.

Magnetic Materials: Introduction - Magnetic dipole moment - Magnetization-Magnetic susceptibility and permeability - Classification of magneticmaterials: Dia, para, Ferro, antiferro & Ferri magnetic materials - Domain concept for Ferromagnetism (Qualitative) - Hysteresis - soft and hard magnetic materials.

Unit Outcomes:

The students will be able to

- **Explain** the concept of dielectric constant and polarization in dielectric materials (L2)
- **Summarize** various types of polarization of dielectrics (L2)
- **Interpret** Lorentz field and Claussius- Mosotti relation in dielectrics(L2)
- **Classify** the magnetic materials based on susceptibility and their temperature dependence(L2)

Unit-IV: Quantum Mechanics and Free electron theory **10hrs**

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 and its temperature dependence.

Unit Outcomes:

The students will be able to

- **Explain** the concept of dual nature of matter (L2)
- **Understand** the significance of wave function (L2)
- **Interpret** the concepts of classical and quantum free electron theories (L2)

Unit – V: Semiconductors **10hrs**

Semiconductors: Formation of energy bands – classification of crystalline solids **-** Intrinsic semiconductors: Density of charge carriers – Electrical conductivity – Extrinsic semiconductors: density of charge carriers **-** Drift and diffusion currents – Einstein's equation **-** Hall effect and its Applications.

Unit Outcomes:

The students will be able to

- **Outline** the properties of charge carriers in semiconductors (L2)
- **Understand** the carrier transportation in semiconductors (L2)
- **Identify** the type of semiconductor using Hall effect (L2)

Text books:

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

Reference Books:

1. "Fundamentals of Physics" - Halliday, Resnick and Walker, John Wiley &Sons.
2. "Engineering Physics" - M.R. Srinivasan, New Age international publishers (2009).
3. "Engineering Physics" - Shatendra Sharma, Jyotsna Sharma, Pearson Education, 2018.
4. "Engineering Physics" - Sanjay D. Jain, D. Sahasrabudhe and Girish, University Press.
5. "Semiconductor physics and devices:Basic principle" - A. Donald, Neamen, Mc GrawHill.
6. "Engineering Physics" - B.K. Pandey and S. Chaturvedi, Cengage Learning
7. "Solid state physics" – A.J.Dekker ,Pan Macmillan publishers
8. "Introduction to Solid State Physics" -Charles Kittel ,Wiley

IYear-I Semester

L	T	P	C
2	0	0	2

COMMUNICATIVE ENGLISH

(Common to All Branches of Engineering)

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

- By the end of the course the students will have Learned how to understand the context, topic, and specific information from social or transactional dialogues.
- Remedially learn applying grammatical structures to formulate sentence sand use appropriate words and correct word forms.
- Using discourse markers to speak clearly on a specific topic in formal as well as informal discussions.(not required)
- Improved communicative competence in formal and informal contexts and for social and academic purposes.
- Critically comprehending and appreciating reading /listening texts and to write summaries.
- Writing coherent paragraphs essays, letters/e-mails and resume.

Instructions:

1. The reading texts can be given as podcasts to the students so that their listening skills can be enhanced.
2. While listening and reading to the text can be given as homework, the class work for the students can be to discuss and critically evaluate the texts based on the context, purpose or writing the text and understanding it from the author's as well as reader's point of view.
3. Reading as habit for both academic and non-academic (pleasure) purposes has to be inculcated in the students. So training has to be given in intensive and extensive reading strategies.
4. Writing for both academic (assignments, examinations, reports, e-mails/letters etc)
5. The writing tasks given in the class are to be self and peer evaluated by the students before they are finally graded by the faculty. Note: Please note that the texts given here are just contexts for teaching various language skills and sub skills. The students' ability to use language cannot be confined to comprehending or using the language related to the given texts (textbooks). The given texts can be used only for practice.
6. All the activities to develop language skills have to be integrated and interconnected, within each unit and across the units.

UNIT I

Lesson: HUMAN VALUES: A Power of a Plate of Rice by Ifeoma Okoye (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.(That has to be part of the bridge course- 2 weeks before the actual academic Programme starts)
Grammar: Parts of Speech, Basic Sentence Structures-forming questions
Vocabulary: Synonyms, Antonyms, Affixes (Prefixes/Suffixes), Root words

UNIT-II

Lesson: **NATURE: Night of the Scorpion by Nissim Ezekiel (Indian and contemporary)**
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 helptolink the ideas in a paragraph together.
Writing: Structure of a paragraph - Paragraph writing (specific topics)
Grammar: Cohesive devices -linkers, useof articles and zeroarticle prepositions.
Vocabulary: Homonyms, Homophones, Homographs.

UNIT-III

Lesson: BIOGRAPHY: Steve Jobs.
Listening: Listening for globalcomprehensionandsummarizingwhatislistenedto.
Speaking: Discussing specific topicsin pairs orsmall groups and reporting what is discussed
Reading: Readingatextindetailbymakingbasicinferences-recognizingandinterpretingspecific Context clues; strategies touse textclues for comprehension.
Writing: Summarizing, Note-making, paraphrasing.
Grammar: Verbs - tenses; subject-verbagreement; 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 elementsintextstoconveyinformation,revealtrends/ Patterns/ relationships, communicate processesor display complicated data.
Writing: Letter Writing: Official Letters, Resumes
Grammar: Reporting verbs, Direct & Indirect speech, Active& PassiveVoice

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, OrientBlackSwan, 2023 (Units 1,2 & 3)
2. Empowering English 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, 2014.
3. Murphy, Raymond. English Grammar in Use, Fourth Edition, Cambridge University Press, 2019.
4. Lewis, Norman. Word Power Made Easy- The Complete Handbook for Building a Superior Vocabulary. Anchor, 2014.

Web Resources:

GRAMMAR:

1. www.bbc.co.uk/learningenglish
2. https://dictionary.cambridge.org/grammar/british-grammar/
3. 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

IYear- ISemester

L	T	P	C
3	0	0	3

BASICCIVIL&MECHANICALENGINEERING
(Common toAll branches of Engineering)

Course Objectives:

- GetfamiliarizedwiththescopeandimportanceofCivilEngineeringsub-divisions.
- Introduce the preliminary conceptsofsurveying.
- Acquire preliminary knowledge on Transportation and its importance innation's economy.
- Get familiarizedwiththeimportanceofquality,conveyanceandstorageofwater.
- Introductiontobasiccivilengineeringmaterialsandconstructiontechniques.

Course Outcomes: On completion of the course,the student should be ableto:

CO1:Understand varioussub-divisionsofCivilEngineeringandtoappreciatetheirroleinensuringbettersociety.

CO2: Know the concepts of surveying and to understand the measurement of distances, angles and levels through surveying.

CO3:Realize the importance of Transportation Innation's economy and the engineering measures related to Transportation.

CO4:Understand the importance of Water Storage and Conveyance Structures so that the social responsibilities of water conservation will be appreciated.

CO5:Understand the basic characteristics of Civil Engineering Materials and attain knowledge on prefabricated technology.

PART A: BASIC CIVIL ENGINEERING

UNITI

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 eachdiscipline-BuildingConstructionandPlanning-ConstructionMaterials-Cement -Aggregate-Bricks-Cementconcrete-Steel.IntroductiontoPrefabricated construction Techniques.

UNITII

Surveying: ObjectivesofSurveying-HorizontalMeasurements-AngularMeasurements-IntroductiontoBearingsLevellinginstrumentsusedforlevelling-Simpleproblemsonlevellingandbearings-Contour mapping.

UNITIII

TransportationEngineeringImportanceofTransportationinNation'seconomicdevelopment- 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.FourthEdition.
2. IntroductiontoCivilEngineering,S.S.Bhavikatti,NewAgeInternationalPublishers.2022.FirstEdition.
3. BasicCivilEngineering,SatheeshGopi,PearsonPublications,2009,FirstEdition.

ReferenceBooks:

1. Surveying,Vol-IandVol-II,S.K.Duggal,TataMcGrawHillPublishers2019.FifthEdition.
2. Hydrology and Water Resources Engineering, Santosh Kumar Garg, KhannaPublishers,Delhi.2016
3. IrrigationEngineeringandHydraulicStructures-SantoshKumarGarg,KhannaPublishers,Delhi2023.38th Edition.
4. HighwayEngineering,S.K.Khanna,C.E.G.JustoandVeeraraghavan,NemchandandBrothers Publications2019. 10th Edition.
5. IndianStandardDRINKINGWATER—SPECIFICATIONIS10500-2012.

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: Understand the different manufacturing processes.
CO2: Explain the basics of thermal engineering and its applications.
CO3: Describe the working of different mechanical power transmission systems and power plants.
CO4:Describe the basics of robotics and its applications.
CO5: Explain the basics of engineering materials and its applications

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

I Year-I Semester

L	T	P	C
3	0	0	3

INTRODUCTION TO PROGRAMMING
(Common to All branches of Engineering)

Course Objectives:

The objectives of this course are to acquire knowledge on the

- To impart adequate knowledge on the need of programming languages and problem-solvingtechniques and develop
  programming skills.
- To enable practical usage of Control Structures and Implement different operations on arrays.
- To demonstrate the use of Strings and Functions.
- To impart the knowledge of pointers and understand the principles of dynamic memoryallocation.
- To understand structures and unions and illustrate the file concepts and its operations.
- To impart the Knowledge Searching and Sorting Techniques.

UNIT-I Introduction to Computer Problem Solving:

Programs and Algorithms, Computer Problem-Solving Requirements, Phases of Problem-Solving, Problem-Solving Strategies, Top-Down Approach, Algorithm Designing, Program Verification, Improving Efficiency, Algorithm Analysis and Notations.

UNIT-II Introduction to C Programming:

Introduction, Structure of a C Program, Comments, Keywords, Identifiers, Data Types, Variables, Constants, Input/Output Statements. Operators, Type Conversion, Control Flow, Relational Expressions: Conditional Branching Statements: if, if-else, if-else—if, switch. BasicLoop Structures: while, do-while loops, for loops, nested loops, The Break and Continue Statements, and goto statements.

UNIT-III Arrays:

Introduction, Operations on Arrays, Arrays as Function Arguments, Two-Dimensional Arrays, Multidimensional Arrays

Pointers: Concept of a Pointer, Declaring and Initializing Pointer Variables, Pointer Expressions and Address Arithmetic, Null Pointers, Generic Pointers, Pointers as Function Arguments, Pointers and Arrays, Pointer to Pointer, Dynamic Memory Allocation, Dangling Pointer, Command Line Arguments.

UNIT-IV Functions:

Introduction Function: Declaration, Function Definition, Function Call, Categories of Functions, Passing Parameters to Functions, Scope of Variables, Variable Storage Classes, Recursion.

Strings: String Fundamentals, String Processing with and without Library Functions, Pointers and Strings.

UNIT-V

Structures, Unions, Bit Fields: Introduction, Nested Structures, Arrays of Structures, Structures and Functions, Self- Referential Structures, Unions, Enumerated Data Type —Enum variables, Using Type def keyword, Bit Fields. Data Files:Introduction to Files, Using Files in C, Reading from Text Files, Writing to Text Files, Random File Access.

Note: The syllabus is designed with C Language as the fundamental language of implementation**.**

Course Outcomes:

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

i. Illustrate the Fundamental concepts of Computers and basics of computer programming andproblem-solving approach.

ii. Understand the Control Structures, branching, and looping statements.

iii. Use of Arrays and Pointers in solving complex problems.

iv. Develop Modular program aspects and Strings fundamentals.

v. Demonstrate the ideas of User Defined Data types files. Solve real-world problems using the concept of Structures,
Unions, and File operations.

Text Books:

1. A Structured Programming Approach Using C, Forouzan, Gilberg, 3rd Edition, Cengage.
2. How to solve it by Computer, R. G. Dromey, 12th Edition, Pearson Education.
3. Programming In C A-Practia1 Approach. Ajay Mittal, 1st Edition Pearson
4. The C Programming Language, Dennis Richie And Brian Kernighan, 2nd Edition, Pearson Education.

References:

1. Byron Gottfried, Schaum's Outline of Programming with C, 4th Edition, 2020, McGraw-Hill.
2. Computer Programming. Reema Thareja, 3rd Edition, 2023, Oxford University Press
3. Computing fundamentals and C Programming, Balagurusamy, E., McGraw-Hill Education, 2008
4. Programming In C, Ashok Kamthane, 2nd Edition, Pearson Publication.

5. Let us C , Yaswanth Kanetkar, 16th Edition, BPB Publication.

Web References:

1. http://www.c4learn.com/
2. http://www.geeksforgeeks.org/c/
3. http://nptel.ac.in/courses/122104019/
4. http://www.learn-c.org/
5. https://www.tutorialspoint.com/cprogramming/

IYear-I Semester

L	T	P	C
0	0	2	1

COMMUNICATIVE ENGLISH LAB

(CommontoAllBranchesofEngineering)

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 students will get trained in the basic communication skills and also make them ready to face job interviews.

Course Outcomes:

- Understand the different aspects of the English language proficiency with emphasis on LSRW skills.
- Apply communication skills through various language learning activities.
- Analyze the English speech sounds, stress, rhythm, intonation and syllabledivision for better listening and speaking comprehension.
- Evaluate and exhibit professionalism in participating in debates and group discussions.
- Create effective resonate and prepare themselves to face interviews in future.

List of Topics:

1. Vowels & Consonants (Not rules but use of them in various syllable structures)
2. Neutralization/Accent Rules (No rules again, required more practice)
3. Communication Skills & JAM
4. Role Play or Conversational Practice
5. (This can be part of theory course)Resume Writing, Cover letter, SOP
6. Group Discussions-methods & practice
7. Debates- Methods & Practice
8. PPT Presentations/ Poster Presentation
9. Interviews Skills

Suggested Software:

- Walden Infotech
- Young India Films

Reference Books:

1.Meenakshi Raman, Sangeeta-Sharma. Technical Communication. Oxford Press.2018.(This can be for theory and not for lab)
2. Samson T : Innovate with English, Foundations
3. Grant Taylor: English Conversation Practice, Tata McGraw-Hill EducationIndia,2016
4. Jayashree, M Let's Hear them Speak: Developing Listening-Speaking skills in English.
. Sage Publications
5. Hewing's, Martin. Cambridge Academic English (B2). CUP, 2012. (That is for reading and writing and can be used in theory classes but not in Lab)
6. T.Balasubramanyam, A Textbook of English Phonetics for Indian Students,(3rd Ed)
Trinity Press. (This is all theory and can be for MA English students but not for B.Tech students)

Web Resources:

Spoken English:

1. www.esl-lab.com
2. www.englishmedialab.com
3. www.englishinteractive.net
4. https://www.britishcouncil.in/english/online
5. http://www.letstalkpodcast.com/
6. 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
12. https://www.linguahouse.com/en-GB
13. https://www.ted.com/watch/ted-ed

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
4. https://www.youtube.com/channel/UCNfm92h83W2i2ijc5Xwp_IA

IYear-I Semester

L	T	P	C
0	0	2	1

ENGINEERING PHYSICS LAB
(Common to All Branches of Engineering)

(Any **TEN** of the following listed experiments)

(Out of which any **TWO** experiments m ay be conducted in virtual mode)

List of Engineering Physics 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 usingdiffraction grating in normal incidence configuration.
3. Verification of Brewster's law
4. Determination of dielectric constant using parallel plate capacitor.
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 energy gap of a semiconductor using PN junction diode.

10 Magnetic field along the axis of a current carrying circular coil by Stewart & Gee'sMethod.

11. Determination of Hall voltage and Hall coefficient of a given semiconductor using Halleffect.
12. Determination of rigidity modulus of the material of the given wire using Torsionalpendulum.
13. V-I Characteristics ofa PN junction diode
14. V-I Characteristics of Zener diode
15. To study the various types of crystal structures.

References:

1. "A Text Book of Practical Physics" - S. Balasubramanian, M.N. Srinivasan, S. ChandPublishers, 2017.

URL:www.vlab.co.in

IYear-I Semester

L	T	P	C
0	0	3	1.5

ENGINEERING WORKSHOP

(Common to All branches of Engineering)

CourseObjectives:

To familiarize students with wood working, sheet metal operations, fitting and electrical housewiring skills

Course Outcomes:

CO1: Identify workshop tools and their operational capabilities.
CO2: Practice on manufacturing of components using workshop trades including fitting, carpentry, foundry and welding.
CO3: Apply fitting operations in various applications.
CO4: Apply basic electrical engineering knowledge for house wiring Practice
CO5: Construct the sheet metal jobs from GI sheets and preparation of pipe joints using plumbing

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 followingfitting 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.

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	C
0	0	2	1

I Year-I Semester

IT WORKSHOP
(Common to all branches of Engineering)

Course Objectives:

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

PC Hardware & Software Installation

Task 1: Identify the peripherals of a computer, components in a CPU, and 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. Labinstructors should verify thework and follow it up with a Viva. Also, students must go through the video showing the PC assembling process. A videowould be given as part of the course content.

Task 3: Students should install MS windows on their personal computer. The labinstructor should verify the installation and follow it 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 Windows and Linux. Lab instructors should verify the installation andfollow it up with a Viva.

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

Internet & World Wide Web

Task 1: Orientation & Connectivity Boot Camp: Students should connect 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. Without internet connectivity, instructors must 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 howto 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 theinstructors by the student.

Task 4: Cyber Hygiene: Students would be exposed to the various threats on the internet and asked to configure their computers to be safe on the internet. They need to customize their browsers to block pop-ups, and block active X downloadsto avoid viruses and worms.

LaTeX and WORD

Task 1 – Word Orientation: The mentor needs to give an overview of LaTeX 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 LaTeX and word – Accessing, overview of toolbars, saving files, Usinghelp 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 options in 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 Contents, 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 the 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 an overview oftoolbars, saving Excel files, Using helpand 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, Hyperlinking, Count function,

LOOKUP/LOOKUP

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

POWERPOINT

Task 1: Students will be working on essential PowerPoint utilities and tools which help them create introductory PowerPoint 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 – Chat GPT

Task 1: Prompt Engineering: Experiment with different prompts to see how the model responds. Try asking questions,

starting conversations, or even providing incomplete sentencesto see how the model completes them.

- Ex: Prompt: "You are a knowledgeable AI. Please answer the following question: Whatis the capital of France?"

Task 2: Creative Writing: Use the model as a writing assistant. Provide the beginning of a storyor a scene description, and

let the model generate the rest of the content. This can be a funway 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 themodel to translate it into another language. Compare the output tosee how accurate and fluent the translations are.

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

Course Outcomes:

CO1: Perform Hardware troubleshooting.

CO2: Understand Hardware components and inter dependencies.

CO3: Safeguard computer systems from viruses/worms.

CO4: Document/ Presentation preparation.

CO5: Perform calculations using spreadsheets.

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, PearsonEducation, 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

I Year-I Semester

L	T	P	C
0	0	3	1.5

COMPUTER PROGRAMMING LAB
(Common to All branches of Engineering)

Course Objectives:

The course aims to give students hands – on experience and train them on the concepts of theC- programming language.

UNIT-I WEEK 1:

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

Suggested Experiments/Activities:

Tutorial 1: Problem-solving using Computers.

Lab1: Familiarization with the 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 seriesof finite steps using textual and
graphic notation.

Suggested Experiments /Activities:

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

Lab 1: 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 traveled 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:

Tutorial4: Operators and the precedence and as associativity:

Lab4: Simple computational problems using the operator's 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. c. A+++B---A
   d. d. J= (i++) + (++i)

ii) Find the maximum of three numbers using the 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 can be used and how to use them. Explore all relational and logical operatorswhile 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 a switch case.

v) Write a C program to find whether the given year is a leap year.

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 eachof these statements is more appropriate.

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 a given number using any loop.

ii) Find whether the given number is a prime or not.

iii) Compute sine and cos series

iv) Checking whether a number is palindrome

v) Construct a pyramid of numbers.

UNIT-III WEEK

7:

Objective: Explore the full scope of the 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 the 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 characters and get comfortable with strings 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-IVWEEK

9:

Objective: Explore pointers to manage a dynamic array of integers, including memory allocation value initialization,
resizing, changing, and reordering the contents of an array, and memory de-allocation using malloc (), calloc (), realloc () and free () functions. Gainexperience 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 self-referential structures(Singly-linked lists), and nested
structures

Suggested Experiments/Activities:

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

Lab10: Bitfields, linked lists

Read and print a date using dd/mm/yyyy format using bit-fields and differentiate the samewithout using bit-fields

i) Create and display a singly linked list using self-referential structure.

ii) Demonstrate the differences between structures and unions using a C program.

iii) Write a C program to shift/rotate using bitfields.

iv) 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 Eulerstheorem.

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 functionsthat can be invoked from themain 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 betweentext 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.

Course Outcomes:

CO1: Read, understand, and trace the execution of programs written in C language.

CO2: Select the proper control structure for solving the problem.

CO3: Develop C programs that utilize memory efficiently using programming constructs likepointers.

CO4: Develop, Debug, and Execute programs to demonstrate the applications of arrays,functions, and basic concepts of
pointers in C.

Text books:

1. Ajay Mittal, Programming in C: A practical approach, 1st Edition, Pearson.
2. Byron Gottfried, Schaum' s Outline of Programming with C, 4th Edition, 2020, McGraw Hill.

Reference Books:

1. Brian W. Kernighan and Dennis M. Ritchie, The C Programming Language, Prentice- Hall of India.

C Programming, A Problem-Solving Approach, Forouzan, Gilberg, Prasad, 3rd Edition, CENGAGE

IYear-ISemester

L	T	P	C
0	0	1	0.5

HEALTH AND WELLNESS, YOGA AND SPORTS

(Common to All branches of Engineering)

Course Objectives:

The main objective of introducing this course is to make the students maintain their mental andphysical wellness by balancing emotions in their life. It mainly enhances the essential traits required for the development of the personality.

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

CO1: Understand the importance of yoga and sports for Physical fitness and sound health.
CO2: Demonstrate an understanding of health-related fitness components. **CO3:** Compare and contrast various activities that help enhance their health.**CO4:** Assess current personal fitness levels.
CO5: Develop Positive Personality

UNIT I

Concept of health and fitness, Nutrition and Balanced diet, basic concept of immunity Relationship between diet and fitness, Globalization and its impact on health, Body Mass Index (BMI) of all age groups.

Activities:

i) Organizing health awareness programmes in community
ii) Preparation of health profile
iii) Preparation of chart for balance diet for all age groups

UNIT II

Concept of yoga, need for and importance of yoga, origin and history of yoga in Indian context, classification of yoga, Physiological effects of Asanas- Pranayama and meditation, stress management and yoga, Mental health and yoga practice.

Activities:

Yoga practices – Asana, Kriya, Mudra, Bandha, Dhyana, Surya Namaskar

UNIT III

Concept of Sports and fitness, importance, fitness components, history of sports, Ancient andModern Olympics, Asian games and Common wealth games.

Activities:

i) Participation in one major game and one individual sport viz., Athletics, Volleyball, Basketball, Handball, Football, Badminton, Kabaddi, Kho-kho, Table tennis, Cricket etc. Practicing general and specific warm up, aerobics
ii) Practicing cardiorespiratory fitness, treadmill, run test, 9 min walk, skipping andrunning.

Reference Books:

1. Gordon Edlin, Eric Golanty. Health and Wellness, 14th Edn. Jones & Bartlett Learning,2022
2. T.K.V.Desikachar. The Heart of Yoga: Developing a Personal Practice
3. Archie J.Bahm. Yoga Sutras of Patanjali, Jain Publishing Company, 1993
4. Wiseman, John Lofty, SAS Survival Handbook: The Ultimate Guide to Surviving Anywhere Third Edition, William Morrow Paperbacks, 2014
5. The Sports Rules Book/ Human Kinetics with Thomas Hanlon. -- 3rd ed. Human Kinetics, Inc.2014

General Guidelines:

1. Institutes must assign slots in the Timetable for the activities of Health/Sports/Yoga.
2. Institutes must provide field/facility and offer the minimum of five choices of as manyas Games/Sports.
3. Institutes are required to provide sports instructor / yoga teacher to mentor the students.

Evaluation Guidelines:

- Evaluated for a total of 100 marks.
- A student can select 6 activities of his/her choice with a minimum of 01 activity per unit. Each activity shall be evaluated by the concerned teacher for 15 marks, totalling to 90 marks.
- A student shall be evaluated by the concerned teacher for 10 marks by conducting vivavoce on the subject.

IYear-II Semester

L	T	P	C
3	0	0	3

DIFFERENTIAL EQUATIONS AND VECTOR CALCULUS
(Commonforallbranches)

Course Objectives:

- To enlighten the learner, sin 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

- Solve the differential equations related to various engineering fields.
- Identify solution methods for partial differential equations that model physical processes.
- Interpret the physical meaning of different operators such as gradient, curl and divergence.
- Estimate the work done against a field, circulation and flux using vector calculus.

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: orthogonal trajectories

Newton's Law of cooling – Law of natural growth and decay- Electrical circuits (RL & RC)

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. Charpt's method Homogeneous & Non-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, del applied to vector point functions-Divergence and Curl, vector identities.

UNIT V Vector integration

Line integral-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 applications of these theorems.

Textbooks:

1. Erwin Kreyszig, Advanced Engineering Mathematics, 10/e, John Wiley & Sons, 2018.
2. B.S. Grewal, Higher EngineeringMathematics,44/e, Khanna publishers,2017.

Reference Books:

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

L	T	P	C
3	0	0	3

IYear-II Semester

Chemistry
Common to EEE, ECE, CSE & IT)

Course Objectives:

- To familiarize engineering chemistry and its applications
- To train the students on the principles and applications of electrochemistry and polymers
- To introduce instrumental methods, molecular machines and switches

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

- Explain the preparation, properties, and applications of thermoplastics & thermosetting, elastomers & conducting polymers.
- Compare the materials of construction for battery and electrochemical sensors.
- Synthesis and characterization of modern engineering materials.
- Explain the principles of spectrometry, Chromatographic separation of solid and liquid mixtures.
- Summarize the conceptsofcolloids,micelle and nanomaterials.

UNIT I Polymer Chemistry

Introduction to polymers, functionality of monomers, chain growth and step growth polymerization,

Plastics –Thermoand 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 II 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-oxygenfuel cell– working of the cells.

UNIT III Modern Engineering materials

Semiconductors, band diagram in solids, Semiconductor devices (p-n junction diode as rectifier and transistors)

Super conductors - Introduction basic concept, applications.

Super capacitors: Introduction, Basic Concept-Classification – Applications.

UNIT IV Instrumental Methods and Applications

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

UNITV Surface Chemistry and Nanomaterials

Introductiontosurfacechemistry,colloids,nanometalsandnanometaloxides,micelleformation,synthesis of

colloids(Bragg'sMethod),Nano materials: Introduction, classification, properties and applications of Fullerenes, carbon nano tubes and Graphines nano particles and metal oxides, BET equation(no derivation)applications of colloid sand nano materials– catalysis, medicine, sensors, etc.

Textbooks:

1. Jain and Jain, Engineering Chemistry, 16/e, DhanpatRai, 2013.
2. Sashi Chawla, Engineering chemistry,Dhanpat rai Publicating Co(Latest edition)

Reference Books:

1. Skoog and West, Principles of Instrumental Analysis, 6/e, Thomson, 2007.
2. J.M.Lehn, Supra Molecular Chemistry, VCH Publications
3. D.J.Shaw,Introduction to Colloids and Surface Chemistry,Butterworth-Heineman,1992.

L	T	P	C
3	0	0	3

IYear- IISemester

ENGINEERING GRAPHICS

(Common to All branches of Engineering)

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

CO1: Understand the principles of engineering drawing, including engineering curves, scales, orthographic and isometric projections.

CO2: Draw and interpret orthographic projections of points, lines, planes and solids in front, top and side views.

CO3: Understand and draw projection of solids in various positions in first quadrant.

CO4: Explain principles behind development of surfaces.

CO5: Prepare isometric and perspective sections of simple solids

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

CO1: Understand the principles of engineering drawing, including engineering curves, scales, orthographic and isometric projections.

CO2: Draw and interpret orthographic projections of points, lines, planes and solids in front, top and side views.

CO3: Understand and draw projection of solids in various positions in first quadrant.

CO4: Explain principles behind development of surfaces.

CO5: Prepare isometric and perspective sections of simple solids

UNIT I

Introduction: Lines, Lettering and Dimensioning, Geometrical Constructions and Constructing regular polygons by general methods.

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

Scales: Plain scales, diagonal scales and vernier scales.

UNIT II

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

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:

1. 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.

IYear-IISemester

L	T	P	C
3	0	0	3

BASICELECTRICAL&ELECTRONICS ENGINEERING
(Common to All branches of Engineering)

PART A: BASIC ELECTRICAL ENGINEERING

CourseObjectives

To expose to the field of electrical engineering, laws and principles of electrical engineeringandto acquirefundamentalknowledgein therelevant field.

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

CO1: Remember the fundamental laws, operating principles of motors, generators, MC and MI instruments.

CO2: Understand the problem-solving concepts associated to AC and DC circuits, construction and operation of AC and DC machines, measuring instruments; different power generation mechanisms, Electricity billing concept and important safety measures related to electrical operations.

CO3: Apply mathematical tools and fundamental concepts to derive various equations related to machines, circuits and measuring instruments; electricity bill calculations and layout representation of electrical power systems.

CO4: Analyze different electrical circuits, performance of machines and measuring instruments.

CO5: Evaluate different circuit configurations, Machine performance and Power systems operation.

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. BasicElectricalEngineering,D.C.Kulshreshtha,TataMcGrawHill,2019,First Edition
2. PowerSystemEngineering,P.V.Gupta,M.L.Soni,U.S.BhatnagarandA.Chakrabarti, Dhanpat Rai& Co, 2013
3. FundamentalsofElectricalEngineering,RajendraPrasad,PHIpublishers,2014,Third Edition

ReferenceBooks:

1. BasicElectricalEngineering,D.P.KothariandI.J.Nagrath,McGrawHill,2019,FourthEdition
2. PrinciplesofPowerSystems,V.K.Mehtha,S.ChandTechnicalPublishers,2020
3. BasicElectricalEngineering,T.K.NagsarkarandM.S.Sukhija,OxfordUniversityPress,2017
4. BasicElectricalandElectronicsEngineering,S.K.Bhatacharya,PersonPublications,2018,Second Edition.

WebResources:

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

PART B: BASIC ELECTRONICS ENGINEERING

CourseObjectives:

To teach the fundamentals of semiconductor devicesand its applications,principles of digitalelectronics.

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

CO1: Remember the fundamental concepts of semiconductor devices, rectifiers, electronic instrumentation systems, and number systems.

CO2: Understand the concepts associated with vacuum tubes, nanoelectronics, diodes, and various configurations and characteristics of transistors and digital electronics.

CO3: Apply mathematical tools and fundamental concepts to derive various equations related to PN diodes, Zener diodes, transistors, and their properties, as well as basic theorems of Boolean algebra.

CO4: Analyze the characteristics of diodes, transistors, rectifiers, and amplifiers, and analyse the truth tables and functionality of logic gates.

CO5: Evaluate different circuit configurations using diodes, transistors, electronic instrumentation systems, simple combinational and sequential circuits, flipflops, registers, and counters.

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 &LouisNashlesky,ElectronicDevices&CircuitTheory,Pearson Education, 2021.
2. R.P.Jain, Modern DigitalElectronics, 4th Edition, TataMcGraw Hill,2009

ReferenceBooks:

1. R.S.Sedha, ATextbook of Electronic DevicesandCircuits, S.Chand&Co,2010.
2. SantiramKal,Basic Electronics-Devices,CircuitsandITFundamentals,PrenticeHall,India,2002.

R.T.Paynter,Introductory ElectronicDevices&Circuits-ConventionalFlowVersion,PearsonEducation, 2009

I Year-II Semester

L	T	P	C
3	0	0	3

DATA STRUCTURES
(Common to CSE, IT & allied branches)

Course Objectives:

- Understand the significance of linear data structures in problem-solving and fundamental time/space complexity analysis.
- Create and manage linked lists to efficiently organize and manipulate data, emphasizing memory efficiency.
- Implement and apply stacks to manage program flow and solve problems involving expression evaluation and backtracking.
- Utilize queues to model real-world scenarios, such as process scheduling and breadth-first search algorithms, understand the versatility of deques, and prioritize data management usingpriority queues.
- Impart a basic understanding of non-linear data structures such as trees.
- Explore basic hashing concepts and apply it to solve problems requiring fast data retrievaland management.

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 ofstacks in expression evaluation, backtracking, reversing lists etc.

UNIT IV

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

Deques: Introduction to deques (double-ended queues), Operations on deques, and theirapplications.

UNIT V

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

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.

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

i) Explain the role of linear data structures in organizing and accessing data efficiently in algorithms.

ii) Design, implement, and apply linked lists for dynamic data storage, demonstrating an understanding of memoryallocation.

iii) Develop programs using stacks to handle recursive algorithms, manage program states, and solve related problems.

iv) Apply queue-based algorithms for efficient task scheduling and breadth-first traversal in graphs and distinguish between deques and priority queues, and apply them appropriately to solve data management challenges.

v) Devise novel solutions to small scale programming challenges involving data structures such asstacks, queues, Trees.

vi) Recognize scenarios where hashing is advantageous, and design hash-based solutions forspecific problems.

Text books:

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 Anderson-Freed,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

IYear-II Semester

L	T	P	C
0	0	2	1

CHEMISTRYLAB

(Common to EEE, ECE, CSE & IT)

. **Course Objectives**:

- Verify the fundamental concepts with experiments.

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

- Determine conductance of solutions.
- Prepare advanced polymer Bakelite materials.
- Measure the strength of an oxidising agent.
- Analyse the IR spectra of some organic compounds.

List of Experiments:

1. Conductometric titration of strong acid vs. strong base
2. Conductometric titration of weak acid vs. strong base
3. Determination of conductance of solutions
4. Determination of strength $KMnO_4$ by using standard oxalic acid solution.
5. Determination of strength of an alkalinity present in water sample.
6. Potentiometry - determination of redox potentials and emfs
7. Preparation of a Bakelite (Demo)
8. Verify Lambert-Beer's law
9. Wavelength measurement of sample through UV-Visible Spectroscopy
10. Measurement of 10Dq by spectrophotometric method
11. Identification of simple organic compounds by IR
12. Preparation of nanomaterials by precipitation method
13. Estimation of Ferrous Iron by Dichrometry.
14. pH metric titration of strong acid vs. strong base

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	C
0	0	3	1.5

IYear-IISemester

ELECTRICAL & ELECTRONICS ENGINEERING WORKSHOP

(Common to All branches of Engineering)

Course Objectives:

To impart knowledge on the fundamental laws & theorems of electrical circuits, functions of electrical machines and energy calculations.

Course Outcomes:

CO1: Understand the Electrical circuit design concept; Measurement of resistance, power, power factor; concept of wiring and operation of Electrical Machines and Transformer.

CO2: Apply the theoretical concepts and operating principles to derive mathematical models for circuits, Electrical machines and measuring instruments; calculations for the measurement of resistance, power and power factor.

CO3: Apply the theoretical concepts to obtain calculations for the measurement of resistance,power and power factor.

CO4: Analyse various characteristics of electrical circuits, electrical machines and measuringinstruments.

CO5: Design suitable circuits and methodologies for the measurement of various electrical parameters; Household and commercial wiring.

Activities:

1. Familiarization of commonly used Electrical & Electronic Workshop Tools: Bread board, Solder, cables, relays, switches, connectors, fuses, Cutter, plier, screwdriver set,wire stripper, flux, knife/blade, soldering iron, de-soldering pump etc.
   - Provide some exercises so that hardware tools and instruments are learned to be used by the students.
2. Familiarization of Measuring Instruments like Voltmeters, Ammeters, Multimeter,LCR-Q meter, Power Supplies, CRO, DSO, Function Generator, Frequency counter.
   - Provide some exercises so that measuring instruments are learned to be used by the students.
3. Components:
   - Familiarization/Identification of components (Resistors, Capacitors, Inductors, Diodes, transistors, IC's etc.) – Functionality, type, size, colour coding package, symbol, cost etc.
   - Testing of components like Resistor, Capacitor, Diode, Transistor, ICs etc. - Compare values of components like resistors, inductors, capacitors etc with the measured values by using instruments

PART A: ELECTRICAL ENGINEERING LAB

List of experiments:

1. Verification of KCL and KVL
2. Verification of Superposition theorem
3. Measurement of Resistance using Wheat stone bridge
4. Magnetization Characteristics of DC shunt Generator
5. Measurement of Power and Power factor using Single-phase wattmeter

6. Measurement of Earth Resistance using Megger
7. Calculation of Electrical Energy for Domestic Premises

Reference Books:

1. Basic Electrical Engineering, D. C. Kulshreshtha, Tata Mc Graw Hill, 2019, FirstEdition
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

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.

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

CO1: Identify & testing of various electronic components.
CO2: Understand the usage of electronic measuring instruments.
CO3: Plot and discuss the characteristics of various electron devices.
CO4: Explain the operation of a digital circuit.
CO5: Realize the truth tables of various Flip flops.

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.

IYear-II Semester

L	T	P	C
0	0	3	1.5

DATA STRCTURES LAB
(Common to CSE, IT & allied branches of Engineering)

Course Objectives:

- Understand the significance of linear data structures in problem-solving and basic time/space complexity analysis.
- Create and manage linked lists to efficiently organize and manipulate data, emphasizing memory efficiency.
- Implement and apply stacks to manage program flow and solve problems involving expression evaluation and backtracking.
- Utilize queues to model real-world scenarios, such as process scheduling and breadth-first search algorithms, understand the versatility of deques, and prioritize data management using priority queues.
- Impart a basic understanding of non-linear data structures such as trees.
- Explore basic hashing concepts and apply them to solve problems requiring fast data retrieval and management.

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 itsproperties 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.

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

i) Explain the role of linear data structures in organizing and accessing data efficiently in algorithms.

ii) Design, implement, and apply linked lists for dynamic data storage, demonstrating an understanding of memoryallocation.

iii) Develop programs using stacks to handle recursive algorithms, manage program states, and solve related problems.

iv) Apply queue-based algorithms for efficient task scheduling and breadth-first traversal in graphs and distinguishbetween deques and priority queues, and apply them appropriately to solve data management challenges.

v) Devise novel solutions to small scale programming challenges involving data structures such asstacks, queues, Trees

vi) Recognize scenarios where hashing is advantageous, and design hash-based solutions forspecific problems.

Text books:

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 Anderson-Freed, SiliconPress, 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.

IYear-II Semester

L	T	P	C
0	0	1	0.5

NSS/NCC/SCOUTS & GUIDES/COMMUNITY SERVICE
(Common to All branches of Engineering)

Course Objectives:
The objective of introducing this course is to impart discipline, character, fraternity, teamwork,social consciousness among the students and engaging them in selfless service.

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

CO1: Understand the importance of discipline, character and service motto.
CO2: Solve some societal issues by applying acquired knowledge, facts, and techniques.
CO3: Explore human relationships by analyzing social problems.
CO4: Determine to extend their help for the fellow beings and downtrodden people.
CO5: Develop leadership skills and civic responsibilities.

UNIT IOrientation
General Orientation on NSS/NCC/ Scouts & Guides/Community Service activities, careerguidance.
Activities:

i) Conducting –ice breaking sessions-expectations from the course-knowing personaltalents and skills
ii) Conducting orientations programs for the students –future plans-activities-releasingroad map etc.
iii) Displaying success stories-motivational biopics- award winning movies on societalissues etc.
iv) Conducting talent show in singing patriotic songs-paintings- any other contribution.

UNIT IINature & Care
Activities:

i) Best out of waste competition.
ii) Poster and signs making competition to spread environmental awareness.
iii) Recycling and environmental pollution article writing competition.
iv) Organising Zero-waste day.
v) Digital Environmental awareness activity via various social media platforms.
vi) Virtual demonstration of different eco-friendly approaches for sustainable living.
vii) Write a summary on any book related to environmental issues.

UNIT III Community Service
Activities:

i) Conducting One Day Special Camp in a village contacting village-area leaders- Surveyin the village, identification of problems- helping them to solve via media- authorities- experts-etc.
ii) Conducting awareness programs on Health-related issues such as General Health,Mental health, Spiritual Health, HIV/AIDS,
iii) Conducting consumer Awareness. Explaining various legal provisions etc.
iv) Women Empowerment Programmes- Sexual Abuse, Adolescent Health and Population Education.
v) Any other programmes in collaboration with local charities, NGOs etc.

Reference Books:

1. Nirmalya Kumar Sinha & Surajit Majumder, A Text Book of National Service Scheme Vol;.I, Vidya Kutir Publication, 2021 (ISBN 978-81-952368-8-6)
2. Red Book - National Cadet Corps – Standing Instructions Vol I & II, Directorate General of NCC, Ministry of Defence, New Delhi
3. Davis M. L. and Cornwell D. A., ―Introduction to Environmental Engineering‖, McGraw Hill, New York 4/e 2008
4. Masters G. M., Joseph K. and Nagendran R. ―Introduction to Environmental Engineering and Science‖, Pearson Education, New Delhi. 2/e 2007
5. Ram Ahuja. Social Problems in India, Rawat Publications, New Delhi.

General Guidelines:

1. Institutes must assign slots in the Timetable for the activities.

2. Institutes are required to provide instructor to mentor the students.

Evaluation Guidelines:

- Evaluated for a total of 100 marks.
- A student can select 6 activities of his/her choice with a minimum of 01 activity per unit. Each activity shall be evaluated by the concerned teacher for 15 marks, totalling to 90 marks.
- A student shall be evaluated by the concerned teacher for 10 marks by conducting vivavoce on the subject.

****

COURSE STRUCTURE
&
DETAILED SYLLABUS

(R23 Regulation)

For

Bachelor of Technology

II B.Tech. (IT)

(Applicable for Batches Admitted from 2023-2024)

Department of

INFORMATION TECHNOLOGY

(Applicable for Batches Admitted from 2023-2024)

DEPARTMENT OF INFORMATION TECHNOLOGY

COLLEGE OF ENGINEERING, VIZIANAGARAM

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY GURAJADA VIZIANAGARAM

ANDHRA PRADESH-535003, INDIA

PROPOSED COURSE STRUCTURE-B.TECH (I.T.) -R23

(Applicable from the academic year 2023-24 onwards)

II-B.Tech I-Semester						
S. No.	**Category**	**Title**	**L**	**T**	**P**	**Credits**
1.	BS & H	Mathematical Foundations of Computer Science	3	0	0	3
2.	BS & H	Universal Human Values - Understanding Harmony	2		0	3
3.	Engineering Science	Digital Logic & Computer Organization	3	0	0	3
4.	Professional Core	Software Engineering	3	0	0	3
5.	Professional Core	Object Oriented Programming Through Java	3	0	0	3
6.	Professional Core	CASE Tools Lab	0	0	3	1.5
7.	Professional Core	Object Oriented Programming Through Java Lab	0	0	3	1.5
8.	Skill Enhancement course	Python Programming	0	1	2	2
9.	Audit Course	Environmental Science	2	0	0	–
		Total	**16**	**2**	**8**	**20**

B.Tech. - II - Year II Semester						
S. No.	**Category**	**Title**	**L**	**T**	**P**	**Credits**
1.	Management Course-I	Managerial Economics and Financial Analysis	2	0	0	2
2.	Engineering Science/ Basic Science	Probability & Statistics	3	0	0	3
3.	Professional Core	Operating Systems	3	0	0	3
4.	Professional Core	Database Management Systems	3	0	0	3
5.	Professional Core	Design and Analysis of Algorithms	3	0	0	3
6.	Professional Core	Operating Systems Lab	0	0	3	1.5
7.	Professional Core	Database Management Systems Lab	0	0	3	1.5
8.	Skill Enhancem ent course	Django Framework	0	1	2	2
9.	BS & H	Design Thinking & Innovation	1	0	2	2
		Total	**15**	**1**	**10**	**21**

II Year I Semester	Mathematical Foundations of Computer Science	L	T	P	C
		3	0	0	3

COURSE OBJECTIVES:

CO1: To understand mathematical arguments using logical connectives and quantifiers and verify the validity of logical flow of arguments using propositional, predicate logic, and truth tables.

CO2: To understand about elementary of combinatorics, the principle of inclusion and exclusion and the pigeonhole principle.

CO3: To expose the students to Binary relations, posets, Hasse diagram, lattice, and discuss various properties of relations.

CO4: To understand Algebraic structures like groups, semigroups, monoids.

CO5: To introduce generating functions and recurrence relations.

COURSE OUTCOMES:

1. Recall the concepts of Mathematical logic and statement & predicate calculus
2. Recall the concepts of combinatorics, set theory, posets and lattices
3. Recall the concepts of algebraic structures, recurrence relations and generating functions
4. Use and interpret the concepts of Mathematical logic and statement & predicate calculus
5. Use and interpret the concepts of combinatorics, set theory, posets and lattices
6. Use and interpret the concepts of algebraic structures, recurrence relations and generating functions
7. Apply the concepts of discreet mathematical structures to computer science and engineering

Unit-I: Mathematical Logic **10 hrs**

Propositional Calculus: Statements and Notations, Connectives, Well Formed Formulas, Truth Tables, Tautologies, and Equivalence of Formulas, Duality Law, Tautological Implications, and Normal Forms, Theory of Inference for Statement Calculus, Consistency of Premises and Indirect Method of Proof.

Predicate Calculus: Predicative Logic, Statement Functions, Variables and Quantifiers, Free and Bound Variables, Inference Theory for Predicate Calculus.

Unit-II: Functions & Relations **10 hrs**

Set Theory: Introduction, Operations on Binary Sets, Principle of Inclusion and Exclusion,

Relations: Properties of Binary Relations, Relation Matrix and Digraph, Operations on Relations, Partition and Covering, Transitive Closure, Equivalence, Compatibility and Partial Ordering Relations, Hassie Diagrams,

Functions: Bijective Functions, Composition of Functions, Inverse Functions, Permutation Functions, Recursive Functions, Lattice and its Properties

Unit-III: Algebraic Structures and Number Theory **10 hrs**

Algebraic Structures: Algebraic Systems, Examples, General Properties, Semi Groups and Monoids, Homomorphism of Semi Groups and Monoids, Group, Subgroup, Abelian Group, Homomorphism, Isomorphism

Number Theory: Properties of Integers, Division Theorem, The Greatest Common Divisor, Euclidean Algorithm, Least Common Multiple, Testing for Prime Numbers, The Fundamental Theorem of Arithmetic, Modular Arithmetic (Fermat's Theorem and Euler's Theorem without Proof)

Unit-IV: Recurrence Relations **8 hrs**

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

Solving Inhomogeneous Recurrence Relations

Unit-V: Graph Theory **10 hrs**

Basic Concepts of Graphs, Sub graphs, Matrix Representation of Graphs: Adjacency Matrices, Incidence Matrices, Isomorphic Graphs, Paths and Circuits, Eulerian and Hamiltonian Graphs, Multigraphs, Planar Graphs, Euler's Formula, Graph Colouring and Covering, Chromatic Number, Spanning Trees, Algorithms for Spanning Trees (Problems Only and Theorems without Proofs)

Text Books:

1. J. P. Tremblay and R. Manohar, Discrete Mathematical Structures with Applications to C Sc, Tata McGraw Hill,1997
2. C. L. Liuand ,Elements of Discrete Mathematics-A Computer Oriented Approach

Reference Books:

1. Kenneth. H. Rosen, Discrete Mathematics and its Applications, 6/e, Tata McGraw-Hill, 2009.
2. Discrete Mathematics for Computer Scientists and Mathematicians, J.L.Mott,A. Kandel, T. P. Baker, 2ndEdition, Prentice Hall of India.
3. Dr. D S Chandrasekharaiah, Mathematical Foundations of Computer Science, Prism Book Pvt Ltd.
4. S. K. Chakraborthy and B.K. Sarkar ,Discrete Mathematics, Oxford,2011

II Year-I Semester	UNIVERSAL HUMAN VALUES – UNDERSTANDING HARMONY AND ETHICAL HUMAN CONDUCT	L	T	P	C
		2	1	0	3

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 behaviour and mutually enriching interaction with Nature.

Course Outcomes:

- Define the terms like Natural Acceptance, Happiness and Prosperity (L1, L2)
- Identify one's self, and one's surroundings (family, society nature) (L1, L2)
- Apply what they have learnt to their own self in different day-to-day settings in real life (L3)
- Relate human values with human relationship and human society. (L4)
- Justify the need for universal human values and harmonious existence (L5)
- Develop as socially and ecologically responsible engineers (L3, L6)

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
Lecture4: 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

READINGS:
Textbook and Teachers Manual
a. 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
b. 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. *JeevanVidya: EkParichaya,* A Nagaraj, JeevanVidyaPrakashan, 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 Beautifu*l - Cecile Andrews
7. *Economy of Permanence* - J C Kumarappa
8. *Bharat Mein Angreji Raj* – PanditSunderlal
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%201%20Teaching%20Material/D3-S2%20Respect%20July%2023.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%203D%20D3-S2A%20Und%20Nature-Existence.pdf
7. https://fdp-si.aicte-india.org/UHV%20II%20Teaching%20Material/UHV%20II%20Lecture%2023-25%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
9. https://onlinecourses.swayam2.ac.in/aic22_ge23/preview

L	T	P	C
3	0	0	3

DIGITAL LOGIC & COMPUTER ORGANIZATION

Course Objectives:

The main objectives of the course is to

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

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 Contra l

UNIT-IV:

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

UNIT-V:

Input/Output Organization: Accessing I/O Devices, Interrupts, Processor Examples, Direct Memory Access, Buses, Interface Circuits, Standard I/O Interfaces

Textbooks:

1. Computer Organization and Embedded Systems, 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, 11thEdition, Pearson.

Reference Books:

1. Computer Systems Architecture, M. Morris 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:

1. https://nptel.ac.in/courses/106/103/106103068/

II Year-I Semester

L	T	P	C
3	0	0	3

SOFTWARE ENGINEERING

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.

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, Software Cost Estimation: 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, McGraw Hill International Edition.

Reference Books:

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

e-Resources:

1) https://nptel.ac.in/courses/I 06/105/106105182/
2) 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

II Year - I Semester

L	T	P	C
3	0	0	3

OBJECT ORIENTED PROGRAMMING THROUGH JAVA

Course Objectives:

The learning objectives of this course are to:

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

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

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

UNIT II

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

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

UNIT III

Arrays: Introduction, Declaration and Initialization of Arrays, Storage of Array m 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-dimensiona1 Arrays, Arrays of Varying Lengths, Three-dimensional Arrays, Arrays as Vectors.

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

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

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

UNIT IV

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

Exception Handling: Introduction, Hierarchy of Standard Exception Classes, Keywords throws and throw, try, catch, and finally Blocks, Multiple Catch Clauses, Class Throw able, Unchecked Exceptions, Checked Exceptions.

Java 1/0 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

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

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

Java FX GUI: 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

II Year - I Semester

L	T	P	C
0	0	3	1.5

CASE TOOLS LAB

COURSE LEVEL LEARNING OUTCOMES (COS)

Students will be able to achieve & demonstrate the following COs on completion of course based learning

Course Objectives

To have hands on experience in developing a software project by using various software engineering principles and methods in each of the phases of software development.

Course Outcomes

1. Ability to translate end-user requirements into system and software requirements
2. Ability to generate a high-level design of the system from the software requirements
3. Will have experience and/or awareness of testing problems and will be able to develop a simple testing report

List of Experiments

Do the following g 8 exercises for any two projects given in the list of sample projects or any other projects:

1) Development of problem statement.
2) Preparation of Software Requirement Specification Document, Design Documents and Testing Phase related documents.
3) Preparation of Software Configuration Management and Risk Management related documents.
4) Study and usage of any Design phase CASE tool
5) Performing the Design by using any Design phase CASE tools.
6) Develop test cases for unit testing and integration testing
7) Develop test cases for various white box and black box testing techniques.

Sample Projects:

1) Passport automation System
2) Book Bank
3) Online Exam Registration
4) Stock Maintenance System
5) Online course reservation system
6) E-ticketing
7) Software Personnel Management System
8) Credit Card Processing
9) E-book management System.
10) Recruitment system

Text Books:

1) Software Engineering, A practitioner's Approach- Roger S. Pressman, 6th edition, Mc Graw Hill International Edition.
2) Software Engineering- Sommerville, 7th edition, Pearson Education.
3) The unified modeling language user guide Grady Booch, James Rambaugh, Ivar Jacobson, Pearson Education.

II Year - I Semester

L	T	P	C
0	0	3	1.5

OBJECT ORIENTED PROGRAMMING THROUGH JAVA 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

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, JavaFX 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 each 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 implement 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 multi level Inheritance

c) Write a JAVA program for abstract class to find areas of different shapes

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) Write a java program that connects to a database using JDBC

b) Write a java program to connect to a database using JDBC and insert values into it.

c) Write a java program to connect to a database using JDBC and delete values from it

II Year - I Semester

L	T	P	C
0	1	2	2

PYTHON PROGRAMMING
(Skill Enhancement Course)

Course Objectives:

- The main objectives of the course are to Introduce core programming concepts of Python programming language.
- Demonstrate about Python data structures like Lists, Tuples, Sets and dictionaries
- Implement Functions, Modules and Regular Expressions in Python Programming and to create practical and contemporary applications using these

UNTI-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:

7. Write a program to define a function with multiple return values.
8. Write a program to define a function using default arguments.
9. Write a program to find the length of the string without using any library functions.

10. Write a program to check if the substring is present in a given string or not.
11. Write a program to perform the given operations on a list:
    i. addition ii. Insertion iii. slicing
12. 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:

13. Write a program to create tuples (name, age, address, college) for at least two members and concatenate the tuples and print the concatenated tuples.
14. Write a program to count the number of vowels in a string (No control flow allowed).
15. Write a program to check if a given key exists in a dictionary or not.
16. Write a program to add a new key-value pair to an existing dictionary.
17. 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:

18. 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.
19. Python program to print each line of a file in reverse order.
20. Python program to compute the number of characters, words and lines in a file.
21. Write a program to create, display, append, insert and reverse the order of the items in the array.
22. Write a program to add, transpose and multiply two matrices.
23. 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 inPython, NumPy with Python, Pandas.

Sample Experiments:

24. Python program to check whether a JSON string contains complex object or not.
25. Python Program to demonstrate NumPy arrays creation using array () function.
26. Python program to demonstrate use of ndim, shape, size, dtype.
27. Python program to demonstrate basic slicing, integer and Boolean indexing.
28. Python program to find min, max, sum, cumulative sum of array

29. 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:
    a) Apply head () function to the pandas data frame
    b) Perform various data selection operations on Data Frame
30. 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. Gowri shankar S, Veena A., Introduction to Python Programming, CRCPress.
2. Python Programming, S Sridhar, J Indumathi, V M Hariharan, 2nd Edition, Pearson, 2024
3. Introduction to Programming Using Python, Y. Daniel Liang, Pearson.

Online Learning Resources/Virtual Labs:

1. https://www.coursera.org/learn/python-for-applied-data-science-ai
2. https://www.coursera.org/learn/python?specialization=python#sylla bus

II Year-I Semester	ENVIRONMENTAL SCIENCE	L	T	P	C
		2	0	0	-

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.

UNIT I **7h**

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 – Water resources – Mineral resources- Food resources–Land resources and Energy resources - Use and over exploitation, case studies.

UNIT II **7h**

Ecosystems: Concept of an ecosystem. – Structure and function of an ecosystem (Producers, consumers and decomposers), Energy flow in the ecosystem and ecological pyramids – Introduction, types, characteristic features of the following ecosystem:

a. Forest ecosystem.
b. Grassland ecosystem
c. Desert ecosystem.
d. Aquatic ecosystems (ponds, streams, lakes, rivers, oceans, estuaries)

Biodiversity and its Conservation: Introduction - Definition: genetic, species and ecosystem diversity – Bio-geographical classification of India – Biodiversity at global, National and local levels – India as a mega-diversity nation – Hot-spots of biodiversity – Threats to biodiversity – Endangered and endemic species of India – Conservation of biodiversity: In-situ and Ex-situ conservation of biodiversity.

UNIT III **6h**

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

a. Air Pollution.
b. Water pollution
c. Soil pollution
d. Marine pollution
e. Noise pollution
f. Thermal pollution
g. Nuclear hazards

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 landslides.

UNIT IV **6h**

Social Issues and the Environment: From Unsustainable to Sustainable development – Water conservation, rain water harvesting, watershed management – Environmental ethics: Issues and possible solutions – Climate change, global warming, acid rain, ozone layer depletion, nuclear accidents and holocaust. Case Studies – Environment Protection Act. – Air (Prevention and Control of Pollution) Act. – Water (Prevention and control of Pollution) Act – Wildlife Protection Act – Forest Conservation Act.

UNIT V **6h**

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, hill slopes, etc.

Textbooks:

1. Textbook of Environmental Studies for Undergraduate Courses Erach Bharucha for University Grants Commission, Universities Press.
2. Palaniswamy, "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.

References:

1. Deeksha Dave and E.Sai Baba Reddy, "Textbook of Environmental Science", Cengage Publications.
2. M.Anji Reddy, "Text book of Environmental Sciences and Technology", BS Publication.
3. J.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 Wendell P. Ela, "Introduction to Environmental Engineering and Science, Prentice hall of India Private limited.

II Year-II Semester	MANAGERIAL ECONOMICS AND FINANCIAL ANALYSIS	L	T	P	C
		2	0	0	2

Course Objectives:

- To inculcate the basic knowledge of microeconomics and financial accounting
- To make the students learn how demand is estimated for different products, input-output relationship for optimizing production and cost
- To Know the Various types of market structure and pricing methods and strategy
- To give an overview on investment appraisal methods to promote the students to learn how to plan long-term investment decisions.
- To provide fundamental skills on accounting and to explain the process of preparing financial statements.

Course Outcomes:

- Define the concepts related to Managerial Economics, financial accounting and management(L2)
- Understand the fundamentals of Economics viz., Demand, Production, cost, revenue and markets (L2)
- Apply the Concept of Production cost and revenues for effective Business decision (L3)
- Analyze how to invest their capital and maximize returns (L4)
- Evaluate the capital budgeting techniques. (L5)
- Develop the accounting statements and evaluate the financial performance of business entity (L5)

UNIT - I Managerial Economics 6h

Introduction – Nature, meaning, significance, functions, and advantages. Demand-Concept, Function, Law of Demand - Demand Elasticity- Types – Measurement. Demand Forecasting- Factors governing Forecasting, Methods. Managerial Economics and Financial Accounting and Management.

UNIT - II Production and Cost Analysis 4h

Introduction – Nature, meaning, significance, functions and advantages. Production Function– Least-cost combination– Short run and long run Production Function- Isoquants and Is costs, Cost & Break-Even Analysis - Cost concepts and Cost behaviour- Break-Even Analysis (BEA) - Determination of Break-Even Point (Simple Problems).

UNIT - III Business Organizations and Markets 4h

Introduction – Forms of Business Organizations- Sole Proprietary - Partnership - Joint Stock Companies - Public Sector Enterprises. Types of Markets - Perfect and Imperfect Competition - Features of Perfect Competition Monopoly- Monopolistic Competition–Oligopoly-Price-Output Determination - Pricing Methods and Strategies

UNIT - IV Capital Budgeting 8h

Introduction – Nature, meaning, significance. Types of Working Capital, Components, Sources of Short-term and Long-term Capital, Estimating Working capital requirements. Capital Budgeting– Features, Proposals, Methods and Evaluation. Projects – Pay Back Method, Accounting Rate of Return (ARR) Net Present Value (NPV) Internal Rate Return (IRR) Method (sample problems)

UNIT - V Financial Accounting and Analysis 10h

Introduction – Concepts and Conventions- Double-Entry Bookkeeping, Journal, Ledger, Trial Balance-Final Accounts (Trading Account, Profit and Loss Account and Balance Sheet with simple adjustments). Introduction to Financial Analysis - Analysis and Interpretation of Liquidity Ratios, Activity Ratios, and

Capital structure Ratios and Profitability.

Textbooks:

1. Varshney & Maheswari: Managerial Economics, Sultan Chand.
2. Aryasri: Business Economics and Financial Analysis, 4/e, MGH.

Reference Books:

1. Ahuja Hl Managerial economics Schand.
2. S.A. Siddiqui and A.S. Siddiqui: Managerial Economics and Financial Analysis, New Age International.
3. Joseph G. Nellis and David Parker: Principles of Business Economics, Pearson, 2/e, New Delhi.
4. Domnick Salvatore: Managerial Economics in a Global Economy, Cengage.

Online Learning Resources:

https://www.slideshare.net/123ps/managerial-economics-ppt
https://www.slideshare.net/rossanz/production-and-cost-45827016
https://www.slideshare.net/darkyla/business-organizations-19917607
https://www.slideshare.net/balarajbl/market-and-classification-of-market
https://www.slideshare.net/ruchi101/capital-budgeting-ppt-59565396
https://www.slideshare.net/ashu1983/financial-accounting

II Year IISemester	PROBABILITY & STATISTICS	L	T	P	C
		3	0	0	3

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

COs	Statements	Blooms level
CO1	Acquire knowledge in finding the analysis of the data quantitatively or categorically and various statistical elementary tools.	L2, L3
CO2	Develop skills in designing mathematical models involving probability, random variables and the critical thinking in the theory of probability and its applications in real life problems.	L3, L5
CO3	Apply the theoretical probability distributions like binomial, Poisson, and Normal in the relevant application areas.	L3
CO4	Analyze to test various hypotheses included in theory and types of errors for large samples.	L2, L3
CO5	Apply the different testing tools like t-test, F-test, chi-square test to analyze the relevant real-life problems.	L3, L5

UNITI:Descriptive statistics and methods for data science: 10 hrs
Data science – Statistics Introduction – Population vs Sample – Collection of data – primary and secondary data – Type of variable: dependent and independent Categorical and Continuous variables– Data visualization – Measures of Central tendency – Measures of Variability (spread or variance) –Skewness Kurtosis.

UNITII:**Probability & Distributions: 10 hrs**
Probability – Conditional probability and Baye's theorem – Random variables – Discrete and Continuous random variables – Distribution function – Mathematical Expectation and Variance –Binomial, Poisson, Uniform and Normal distributions.

UNITIII:**SamplingTheory: 10 hrs**
Introduction – Population and samples – Sampling distribution of Means and Variance (definition only) – Central limit theorem (without proof) – Introduction to t, chi-square and F-distributions – Point and Interval estimations – Standard error and Maximum error of estimate.

UNITIV:Tests of Hypothesis: 10 hrs
Introduction – Hypothesis – Null and Alternative Hypothesis – Type I and Type II errors – Level of significance-Confidence limits-Test of significance for large samples-single and two means – single and two proportions-Student's t- distribution- significance test of a sample mean – significance test of difference between sample means.F-test, chi-square test and test of goodness of fit.

UNITV :Regression analysis: 8 hrs
Method of least squares – Straight line – Parabola – Exponential – Power curves. Regression -Regression coefficients and properties – Curvilinear Regression, Multiple Regression - Correlation –Correlation coefficient – Rank correlation

Textbooks:
1. Miller and Freunds, 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. S. Ross, a First Course in Probability, Pearson Education India, 2002.
3. W. Feller, an Introduction to Probability Theory and its Applications, 1/e, Wiley, 1968.

Online Learning Sources:

- https://onlinecourses.nptel.ac.in/noc21_ma74/preview
- https://onlinecourses.nptel.ac.in/noc22_mg31/preview

II Year - II Semester

L	T	P	C
3	0	0	3

OPERATING SYSTEMS

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.

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, 10th Edition, Wiley, 2018.
2. Modern Operating Systems, Tanenbaum A S, 4th Edition, Pearson ,2016

Reference Books:

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

Online Learning Resources:

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

II Year - II Semester

L	T	P	C
3	0	0	3

DATABASE MANAGEMENT SYSTEMS

Course Objectives:

- The main objectives of the course is to 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

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.

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

UNIT II:

Relational Model: Introduction to relational model, concepts of domain, attribute, tuple, relation, importance of null values, constraints (Domain, Key constraints, integrity constraints) and their importance, Relational Algebra - Selection and Projection, Set Operations, Renaming, Joins, Division, More Examples of Relational Algebra Queries , Relational Calculus- Tuple Relational Calculus, Domain Relational Calculus, Expressive. Power of Algebra and Calculus. BASIC SQL: Simple Database schema, data types, table definitions - DDL (Data Definition Language) - create, alter, different DML operations (insert, delete, update).

UNIT III:

SQL: Basic 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 Concept: Transaction State, ACID properties, Concurrent Executions, Serializability, Recoverability, Implementation of Isolation, Testing for Serializability, lock based, time stamp based, optimistic, concurrency protocols, Deadlocks, Failure Classification, Storage, Recovery and Atomicity, Recovery algorithm.

Introduction to Indexing Techniques: B+ Trees, operations on B+ Trees, 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

II Year - II Semester

L	T	P	C
3	0	0	3

DESIGN & ANALYSIS OF ALGORITHMS

Course Objectives:

The main objectives of the course is to

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

Course Outcomes:

After completion of the course, students will be able to

1. Illustrate the working of the advanced tree data structures and their applications (L2)
2. Understand the Graph data structure, traversals and apply them in various contexts. (L2)
3. Use various data structures in the design of algorithms (L3)
4. Recommend appropriate data structures based on the problem being solved (L5)
5. Analyze algorithms with respect to space and time complexities (L4)

UNIT – I:

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

SETS and DISJOINT SETS – Union and Find Operations.

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

UNIT – II:

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

Divide and Conquer: The General Method, Quick Sort, Merge Sort, Strassen's matrix multiplication, Convex Hull

UNIT – III:

Greedy Method: General Method, Job Sequencing with deadlines, Knapsack Problem, Minimum cost spanning trees, Single Source Shortest Paths

UNIT – IV:

Dynamic Programming: General Method, applications-Matrix chain multiplication, optimal binary search trees,0/1 Knapsack problem, all pairs shortest paths, Travelling Salesperson problem, Reliability Design.

UNIT – V:

Backtracking: General Method, Applications – N-Queens Problem, Sum of Subsets problem, Graph Coloring, Hamiltonian Cycle.

Branch and Bound: The General Method, applications-Travelling Salesperson problem, 0/1 Knapsack Problem, LC Branch and Bound solution, FIFO Branch and Bound solution.

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

Textbooks:

1. Fundamentals of Computer Algorithms, Ellis Horowitz, Satraj Sahni and Rajasekharam, Universities Press.
2. Introduction to Algorithms, second edition, T.H. Cormen, C.E.Leiserson, R.L.Rivest and C.Stein, PHI Pvt.Ltd.
3. Introduction to the Design and Analysis of Algorithms, Anany Levitin, PEA.
4. Algorithm Design, Jon Kleinberg and Eva Tardos. 1st Edition.
5. Data Structures and Algorithms in Python. Roberto Tamassia, Michael H. Goldwasser, Michael T. Goodrich. Wiley.1st Edition.
6. Problem Solving with Algorithms and Data Structures Using Python. Release 3.0. Bradley N Miller. Franklin Beedle & Assoc

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
5. 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
2. http://peterindia.net/Algorithms.html
3. https://www.youtube.com/playlist?list=PLDN4rrl48XKpZkf03iYFl-O29szjTrs_O

II Year - II Semester

L	T	P	C
0	0	3	1.5

OPERATING SYSTEMS LAB

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

Course Outcomes:

After completion of the course, students will be able to

1. Trace different CPU Scheduling algorithms (L2).
2. Implement Bankers Algorithms to Avoid the Dead Lock (L3).
3. Evaluate Page replacement algorithms (L5).
4. Illustrate the file organization techniques (L4).
5. Illustrate Inter process Communication and concurrent execution of threads (L4)

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

Sample Experiments:

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. Android Experiments:
    a) Installation of Android studio.
    b) Development of Hello World Application.
    c) Create an application that takes the name from a text box and shows hello message along with the name entered in text box, when the user clicks the OK button.

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 Dhamdhere, 3rd Edition, McGraw- Hill, 2013

Online Learning Resources:

1. https://www.cse.iitb.ac.in/~mythili/os/
2. http://peterindia.net/OperatingSystems.html

II Year - II Semester

L	T	P	C
0	0	3	1.5

DATABASE MANAGEMENT SYSTEMS LAB

Course Objectives:

This course will enable students to

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

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 droping of tables and inserting rows into a table(use constraints while creating tables) examples using SELECT command.
2. Queries (along with sub Queries) using ANY, ALL, IN, EXISTS, NOTEXISTS, UNION, INTERSET, Constraints. Example:- Select the roll number and name of the student who secured fourth rank in the class.
3. Queries using Aggregate functions (COUNT, SUM, AVG, MAX and MIN), GROUP BY, HAVING and Creation and dropping of Views.
4. Queries using Conversion functions (to_char, to_number and to_date), string functions (Concatenation, lpad, rpad, ltrim, rtrim, lower, upper, initcap, length, substr and instr), date functions (Sysdate, next_day,add_months, last_day, months_between, least, greatest, trunc, round, to_char, to_date)
5. 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 ifno records were found)
   ii. Insert data into student table and use COMMIT, ROLLBACK and SAVEPOINT in PL/SQL block.
6. Develop queries by using NESTED queries concept.
7. Develop a program that includes the features NESTED IF, CASE and CASE expression. The program can be extended using the NULLIF and COALESCE functions.
8. Program development using WHILE LOOPS, numeric FOR LOOPS, nested loops using ERROR Handling, BUILT –IN Exceptions, USE defined Exceptions, RAISE- APPLICATION ERROR.
9. Programs development using creation of procedures, passing parameters IN and OUT of PROCEDURES.

10. Program development using creation of stored functions, invoke functions in SQL Statements and write complex functions.
11. Develop programs using features parameters in a CURSOR, FOR UPDATE CURSOR, WHERE CURRENT of clause and CURSOR variables.
12. Develop Programs using BEFORE and AFTER Triggers, Row and Statement Triggers and INSTEAD OF Triggers
13. Create a table and perform the search operation on table using indexing and non-indexing techniques.
14. Write a Java program that connects to a database using JDBC
15. Write a Java program to connect to a database using JDBC and insert values into it
16. Write a Java program to connect to a database using JDBC and delete values from it

Text Books/Suggested Reading:

1. High Performance MySQL: Proven Strategies for Operating at Scale. Author(s): Silvia Botros, Jeremy Tinley. Publisher: O'Reilly Media, Year: 2021
2. Rick F Vander Lans, "Introduction to SQL", Fourth Edition, Pearson Education, 2007

E-Resource:

Online resource: https://hsqldb.org/

II Year - II Semester

L	T	P	C
0	1	2	2

DJANGO FRAMEWORK
(Skill Enhancement Course)

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

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 / BottleFramework

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: Database Migrations, Fetch Data From Database, Displaying Data On Templates, Adding Condition 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 Static Root, 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 Herokuwith all the functionalities developed.
3. Configure Django to handle static files.

Optional Experiments:

1. Setting Up a Django Project with Fast API Integration.
2. Creating a CRUD API with Fast API in a Django Project.
3. Implementing Authentication with Django and Fast API

Text books:

1. Martin C.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, A., press, "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", 2nd Edition 2018, Packt Publishing.
2. Harry Percival, "Test-Driven Development with Python: Obey the Testing Goat: Using Django, Selenium and JavaScript", 2nd Edition 2019, Kindle Edition.

E-Resources:

1. https://medium.com/@mohanishp1/building-web-applications-with-django-and-fastapi-combining-the-best-of-both-worlds-1892719a8b9d.
2. https://sunscrapers.com/blog/fastapi-and-django-a-guide-to-elegant-integration/

II Year-II Semester	DESIGN THINKING & INNOVATION	L	T	P	C
		1	0	2	2

Course Objectives:
The objective of this course is to familiarize students with design thinking process as a tool for breakthrough innovation. It aims to equip students with design thinking skills and ignite the minds to create innovative ideas, develop solutions for real-time problems.

Course Outcomes:
- Define the concepts related to design thinking. (L1, L2)
- Explain the fundamentals of Design Thinking and innovation (L1, L2)
- Apply the design thinking techniques for solving problems in various sectors. (L3)
- Analyse to work in a multidisciplinary environment (L4)
- Evaluate the value of creativity (L5)
- Formulate specific problem statements of real time issues (L3, L6)

UNIT I Introduction to Design Thinking 10h
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 10h
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 10h
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 8h
Problem formation, introduction to product design, Product strategies, Product value, Product planning, product specifications- Innovation towards product design- Case studies
Activity: Importance of modelling, how to set specifications, Explaining their own product design.

UNIT V Design Thinking in Business Processes 10h
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, Harper Bollins (2009)
2. Idris Mootee, Design Thinking for Strategic Innovation, 2013, John Wiley & Sons.

Reference Books:
1. David Lee, Design Thinking in the Classroom, Ulysses press

2. Shrutin N Shetty, Design the Future, Norton Press
3. William Lidwell, Universal Principles of Design- Kritinaholden, Jill Butter.
4. Chesbrough. H, The Era of Open Innovation – 2013

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

COURSE STRUCTURE
&
DETAILED SYLLABUS
(R23 Regulation)
For
Bachelor of Technology
III B.Tech. (IT) & IV B.Tech.(IT)
B.Tech(Honors & Minors)
(Applicable for Batches Admitted from 2023-2024)

Department of
INFORMATION TECHNOLOGY
(Applicable for Batches Admitted from 2023-2024)

DEPARTMENT OF INFORMATION TECHNOLOGY

COLLEGE OF ENGINEERING, VIZIANAGARAM

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY GURAJADA
VIZIANAGARAM

ANDHRA PRADESH-535003, INDIA

JNTUGV- COLLEGE OF ENGINEERING VIZIANAGARAM

DEPARTMENT OF INFORMATION TECHNOLOGY

B.TECH- IT (R23)

B.Tech. – III Year I Semester

S.No.	Category	Title	L	T	P	Credits
1	Professional Core	Advanced Java	3	0	0	3
2	Professional Core	Computer Networks	3	0	0	3
3	Professional Core	Automata Theory & Compiler Design	3	0	0	3
4	Professional Elective-I	1. Object Oriented Analysis and Design 2. Cyber Security 3. Artificial Intelligence 4. Microprocessors & Microcontrollers 5. Data Warehousing & Data Mining 6. 8/12 week MOOC Swayam/ NPTEL course recommended by the BoS	3	0	0	3
5	Open Elective- I	1. Principles of Operating Systems 2. Computer Organization and Architecture	3	0	0	3
6	Professional Core	Advanced Java Lab	0	0	3	1.5
7	Professional Core	Computer Networks Lab	0	0	3	1.5
8	Skill Enhancement course	Full Stack Development 1	0	1	2	2
9	Engineering Science	User Interface Design using Flutter / SWAYAM Plus - Android Application Development (with Flutter)	0	0	2	1
10	Evaluation of Community Service Internship		-	-	-	2
	Total		15	1	10	23
MC	Minor Course (Student may select from the same specialized minors pool)		3	0	0	3
MC	Minor Course through SWAYAM/NPTEL (minimum 12 week, 3 credit course)		3	0	0	3
HC	Honors Course (Student may select from the same honors pool)		3	0	0	3
HC	Honors Course (Student may select from the same honors pool)		3	0	0	3

B.Tech. III Year II Semester

S.No.	Category	Title	L	T	P	Credits
1	Professional Core	Advanced Data Structures	3	0	0	3
2	Professional Core	Cryptography & Network Security	3	0	0	3
3	Professional Core	Machine Learning	3	0	0	3
4	Professional Elective-II	1. Software Testing Methodologies 2. DevOps 3. Generative AI 4. Intelligent AI Agents 5. 8/12 week MOOC Swayam/NPTEL course recommended by the BoS	3	0	0	3
5	Professional Elective-III	1. Software Project Management 2. Mobile Adhoc Networks 3. Natural Language Processing 4. Distributed Operating System 5. 8/12 week MOOC Swayam/NPTEL course recommended by the BoS	3	0	0	3
6	Open Elective – II	Principles of Database Management Systems	3	0	0	3
7	Professional Core	Advanced Data structures Lab	0	0	3	1.5
8	Professional Core	Machine Learning Lab	0	0	3	1.5
9	Skill Enhancement course	Salesforce AI Agent	0	1	2	2
10	Audit Course	Technical Paper Writing & IPR	2	0	0	-
Total			**20**	**1**	**08**	**23**
Mandatory Industry Internship / Mini Project of 08 weeks duration during summer vacation						

MC	Minor Course (Student may select from the same specialized minors pool)	3	0	3	4.5
HC	Honors Course (Student may select from the same honors pool)	3	0	0	3

* Under Industry Internship interested students can pursue SWAYAM Plus courses viz., Hands-on Masterclass on Data Analytics OR Artificial Intelligence for Real-World Application

B.Tech. IV Year I Semester

S.No.	Category	Title	L	T	P	Credits
1	Professional Core	Internet of Things	3	0	0	3
2	Management Course- II	Human Resources & Project Management	2	0	0	2
3	Professional Elective-IV	1. Software Architecture & Design Pattern 2. Deep Learning 3. Computer Vision 4. Block chain Technology 5. 8/12 week MOOC Swayam/NPTEL course recommended by the BoS	3	0	0	3
4	Professional Elective-V	1. Agile methodologies 2. Big Data Analytics 3. Mobile Computing 4. Cyber Physical Systems 5. 8/12 week MOOC Swayam/NPTEL course recommended by the BoS 6. Cloud Computing 7. Intro to LLM	3	0	0	3
5	Open Elective-III	Object-Oriented Programming Through Java	3	0	0	3
6	Open Elective-IV	1.Principles of Software Engineering 2.Computer Networks	3	0	0	3
7	Skill Enhancement Course	Prompt Engineering/ SWAYAM Plus - Certificate program in Prompt Engineering and ChatGPT	0	1	2	2
8	Audit Course	Constitution of India	2	0	0	-
9	Internship	Evaluation of Industry Internship / Mini Project	-	-	-	2

	Total	**19**	**1**	**02**	**21**
MC	Minor Course (Student may select from the same specialized minors pool)	3	0	0	3
HC	Honors Course (Student may select from the same honors pool)	3	0	0	3
HC	Honors Course (Student may select from the same honors pool)	3	0	0	3

B.Tech. IV Year II Semester

S.No.	Category	Title	L	T	P	Credits
1	Internship & Project Work	Full semester Internship / Project Work	0	0	24	12

Note : ***Students need to do at least ONE MOOC/ NPTEL Course (of 3 credits out of 160 credits) to meet the mandatory requirement (11th criteria, as per R23 Regulations); they are allowed to register one semester in advance***

Open Electives, offered to other department students:

Open Elective I	1. Principles of Operating Systems 2. Computer Organization and Architecture
Open Elective II	Principles of Database Management Systems
Open Elective IIII	Object Oriented Programming Through Java
Open Elective IV	1. Principles of Software Engineering 2. Computer Networks

III Year I Semester	ADVANCED JAVA	L	T	P	C
		3	0	0	3

Course Objectives:

The course aims to:

1. Introduce the architecture and components of J2EE and their role in web development.
2. Familiarize students with servlets and their lifecycle, configuration, and advanced web application handling.
3. Provide a detailed understanding of Java Server Pages (JSP), scripting elements, and web application state management.
4. Enable students to interact with databases using JDBC and build robust, database-driven applications.
5. Introduce the Spring MVC framework and demonstrate how enterprise-level applications can be structured using Spring features such as dependency injection, AOP, and DAO modules.

Course Outcomes:

By the end of the course, students will be able to:

1. **Understand** the architecture of J2EE and the different types of containers and servers used in enterprise web applications.
2. **Develop** web applications using servlets, manage their lifecycle, and implement session tracking mechanisms efficiently.
3. **Design and implement** JSP-based dynamic web pages utilizing scripting, directives, JSTL, and expression language.
4. **Apply** JDBC APIs to interact with relational databases and perform CRUD operations using different statement interfaces and result set operations.
5. **Build** scalable, maintainable enterprise applications using Spring MVC with core features like Bean Factory, Dependency Injection, AOP, and transaction management.

UNIT - I:

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 - II:

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 Servelt,, Response and Redirection using Request Dispacher 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: ContextLevel and SessionLevel.

UNIT - III:

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-IV:

JDBC Programming: JDBC Architecture, Types of JDBC Drivers, Introduction to major JDBC Classes and Interface, Creating simple JDBC Application, Types of Statement (Statement Interface, PreparedStatement, CallableStatement), Exploring ResultSet Operations, Batch Updates in JDBC, Creating CRUD Application, Using Rowsets Objects, Managing Database Transaction.

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. JDBC™ API Tutorial and Reference, Third Edition, Maydene Fisher, Jon Ellis, Jonathan Bruce, Addison Wesley
3. Beginning JSP, JSF and Tomcat, Giulio Zambon, Apress

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

Course Objectives:

The course aims to:

1. Provide a clear understanding of the basic requirements of network hardware, software, and architecture.
2. Familiarize students with the layered architecture of network protocols and the hierarchical structure of physical network infrastructure.
3. Introduce various network interconnecting devices and associated hardware components.
4. Explore advanced networking concepts including wireless and wireless sensor networks.

Course Outcomes (COs):

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

1. Explain the fundamental concepts of computer networks, reference models, and network hardware/software components.
2. Analyze the functionalities of data link and network layers, including error detection, routing algorithms, and IP addressing schemes.
3. Apply appropriate protocols for designing and configuring reliable communication between computers across networks.
4. Evaluate the performance of various transport protocols and network services such as TCP, UDP, DNS, HTTP, and email.
5. Compare and contrast wired and wireless network technologies, devices, and emerging concepts in networking.

UNIT--I:

Introduction to Computer Networks: Introduction, Network Hardware, Network Software, Reference Models,Network Examples, Internet-Based Applications. Architecture of the Internet.

Physical Layer: Guided transmission media, Wireless transmission media, Switching- **Circuit Switching:** Phases:Setup, data transfer, teardown, use in traditional telephone networks, Packet Switching, Store-and-forward, statistical multiplexing, Advantages over circuit switching

UNIT- - II:

Data Link Layer - Design issues, Error Detection & Correction, Elementary Data Link Layer Protocols, Sliding window protocols Multiple Access Protocols - ALOHA, CSMA, CSMA/CD, CSMA/CA, Collision free protocols,

Ethernet- Physical Layer, Ethernet Mac Sub layer,

Data link layer switching: Use of bridges, learning bridges, spanning tree bridges, repeaters, hubs, bridges, switches, routers and gateways.

UNIT- - III:

Network Layer: Network Layer Design issues, store and forward packet switching connection less and connection oriented networks-routing algorithms-optimality principle, shortest path, flooding, Distance Vector Routing, Count to Infinity Problem, Link State Routing, Path Vector Routing, Hierarchical Routing; Congestion control algorithms, IP addresses, CIDR, Subnetting, SuperNetting, IPv4, Packet Fragmentation.

UNIT-- IV:

Transport layer: Transport Service, Elements of Transport Protocols, TCP and UDP Protocols, Quality of Service Model, Best Effort Model, Network Performance Issues.

UNIT- -V:

Application Layer: Overview of DNS, World Wide Web, HTTP Protocols, Electronic Mail, and Firewalls.

Text Books:

1. Computer Networks, Andrew S Tanenbaum. (6th ed.). Pearson Edu.
2. Computer Networking. TOP DOWN APP. Kurose, J. F., & Ross, K. W. (8th ed.). Pearson Education.

References:

1. Data Communications and Networking , Behrouz A Forouzan , Tata McGraw-Hill Co Ltd, Second Edition,
2. ISBN: 0-07-049935-7
3. Computer networks, Mayank Dave, CENGAGE.
4. Computer networks, A system Approach, 5thed, Larry L Peterson and Bruce S Davie, Elsevier.
5. Understanding communications and Networks, 3rd Edition, W.A. Shay, Thomson.

III Year I Semester	AUTOMATA THEORY & COMPILER DESIGN	L	T	P	C
		3	0	0	3

Course Objectives:

1. To introduce the fundamentals of formal languages, grammars, and finite automata.
2. To design and analyze regular expressions, finite automata (FAs), and pushdown automata (PDAs).
3. To develop an understanding of the theoretical foundations and practical approaches used in compiler construction.
4. To explore lexical analysis, parsing strategies, and syntax-directed translation.
5. To familiarize students with intermediate code generation, code optimization techniques, and target code generation.

Course Outcomes :

By the end of the course, students will be able to:

1. Explain and apply formal language theory, including regular expressions, finite automata, and the Chomsky hierarchy.
2. Design context-free grammars and construct pushdown automata for language recognition.
3. Analyze lexical analysis and implement top-down parsing techniques, including LL(1) parsing and recursive descent.
4. Construct bottom-up parsers using LR, SLR, and LALR methods, and apply syntax-directed translation.
5. Generate intermediate and target code, perform type checking, and apply basic code optimization techniques.

UNIT- – I:

Regular Expressions, Languages and Finite Automata - Formal Languages and the Chomsky Hierarchy, Regular Expressions and Regular Languages, Algebraic Laws for Regular Expressions, Applications of Regular Expressions, Abstract model of Finite Automaton, Transition Tables and Transition Graphs, Deterministic Finite Automata (DFA), Nondeterministic Finite Automata (NFA), Converting NFA to DFA, Finite Automata with ϵ transitions (NFA-ϵ), Converting NFA-ϵ to NFA/DFA, Minimization of Finite Automata, Equivalence of FA and Regular Expressions

UNIT--II:

Context Free Grammars and Push Down Automata: Context Free Grammars (CFG) and Context Free Languages (CFL), Design of CFGs, Leftmost and Rightmost Derivations, Parse Trees, Applications of CFGs, Ambiguity in Grammars and

Languages, Push Down Automata (PDA), The Language of a PDA, Equivalence of PDAs and CFGs

UNIT--III:

Lexical Analysis and Top-Down Parsing-The structure of a compiler, Role of lexical analyzer, Input Buffering, Specification of tokens, Recognition of tokens, The Lexical Analyser Generator –LEX ; Introduction to Syntax Analysis, Eliminating ambiguity and left recursion from a CFG, Recursive Decent Parsing, LL(1) Grammars, Nonrecursive Predictive Parsing

UNIT--IV:

Bottom-Up Parsing and Syntax Directed Translation- Shift-Reduce Parsing, Simple LR parsing, Canonical LR(1) Parsing, LALR Parsing, Parser Generators , Syntax Directed Definitions, Evaluation Orders for SDDs, Syntax Directed Translation Schemes

UNIT--V:

Intermediate Code Generation, Code Generation and Optimization: Three address code, Types and Declarations, Translation of Expressions, Type Checking, Control Flow, Issues in the design of a Code Generator, The Target Language, A simple Code Generator Basic Blocks and Flow Graphs, Optimization of Basic Blocks, Peephole Optimization

Textbooks:

1. Introduction to Automata Theory, Languages and Computation, J.E.Hopcroft, R.Motwani and J.D.Ullman, 3rd Edition, Pearson, 2008.
2. Compilers Principles, Techniques and Tools, 2nd Edition, Alfred V.Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman, Pearson

Reference Books:

1. Introduction to Languages and The Theory of Computation, John C. Martin, McGraw Hill.
2. Theory of Computer Science-Automata, Languages and Computation, K.L.P.Mishra and N.Chandrasekaran, 3rd Edition, PHI, 2007
3. Compiler Construction‖, K.V.N. SUNIT-ha, Pearson, 2013
4. Compiler Design, SandeepSaxena, Rajkumar Singh Rathore, S.Chand publication

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

Course Objectives:

1. Become familiar with all phases of Object-Oriented Analysis and Design (OOAD).
2. Master the key features and diagrams of the Unified Modeling Language (UML).
3. Understand and apply object-oriented concepts and technologies to solve real-world problems across various domains.
4. Learn and apply object design principles to guide effective implementation of software systems.

Course Outcomes:

By the end of the course, students will be able to:

1. Understand the structure and complexity of software systems and design solutions using object-oriented principles.
2. Create and interpret UML models, including class, object, and package diagrams, to represent static system structure.
3. Model system behavior using use cases, interaction diagrams, and activity diagrams based on functional requirements.
4. Develop advanced behavioral and architectural models, including state machines, component, and deployment diagrams.
5. Apply OOAD techniques and UML modeling in practical case studies and real-world application domains.

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

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

Course Objectives:

The course aims to:

1. Introduce the fundamental concepts, objectives, and roles in cyber security.
2. Provide understanding of security architecture and lifecycle management in information security.
3. Equip students with knowledge of incident response, operational security measures, and risk management.
4. Familiarize students with threat detection, monitoring tools, and traffic analysis techniques.
5. Introduce backdoor systems, metasploit framework, and methods to secure operating systems.

Course Outcomes:

1. Understand the fundamental principles, objectives, and roles of cyber security, and differentiate it from information security.
2. Apply concepts of security architecture, risk management, and lifecycle management to identify and mitigate risks and vulnerabilities in information systems.
3. Analyze and implement incident response strategies, operational security practices, and configuration management to secure digital assets and network infrastructure.
4. Utilize monitoring tools and techniques for threat detection, log analysis, and network traffic evaluation to maintain security posture.
5. Demonstrate knowledge of backdoor systems, penetration testing tools like Metasploit, and system hardening techniques to enhance system security.

UNIT--I:

Introduction to Cyber security- Cyber security objectives, Cyber security roles, Differences between Information Security & Cyber security, Cyber security Principles Confidentiality, integrity, &availability Authentication & non- repudiation.

UNIT--II:

Information Security (IS) within Lifecycle Management-Lifecycle management landscape, Security architecture processes, Security architecture tools, Intermediate

lifecycle management concepts, Risks & Vulnerabilities-Basics of risk management, Operational threat environments, Classes of attacks.

UNIT--III:

Incident Response- Incident categories, Incident response Incident recovery, and Operational security protection: Digital and data assets, ports and protocols, Protection technologies, Identity and access Management, configuration management.

UNIT--IV:

Threat Detection and Evaluation (DE): Monitoring- Vulnerability Management, Security Logs and Alerts, Monitoring Tools and Appliances. Analysis- Network traffic Analysis, packet capture and analysis

UNIT--V:

Introduction to backdoor System and security-Introduction to metasploit, Backdoor, demilitarized zone(DMZ),Digital Signature, Brief study on Harding of operating system.

Text Books:

1. NASSCOM: Security Analyst Student Hand Book.

2. Information Security Management Principles Updated Edition by David Alexander, Amanda Finch, David Sutton ,Published by BCS.

Reference Books:

1. CSX- cyber security fundamentals , Published by ISACA, Cyber security, Network Security, Data Governance Security.

E-Resources:

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

III Year I Semester	ARTIFICIAL INTELLIGENCE (Professional Elective-I)	L	T	P	C
		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. To introduce students to the fundamental concepts and goals of Artificial Intelligence.
2. To teach various AI techniques for solving complex and real-world problems.
3. To familiarize students with the structure and functioning of Expert Systems.
4. To explore AI applications such as game playing, machine learning, and automated theorem proving.
5. To understand and apply different knowledge representation and reasoning techniques.

Course Outcomes :

By the end of the course, students will be able to:

1. Explain the foundational principles and history of Artificial Intelligence, and describe intelligent agents and their environments.
2. Apply various search strategies (uninformed and informed) and adversarial search techniques like minimax and alpha-beta pruning.
3. Represent and reason with knowledge using predicate logic, semantic networks, rules, and probabilistic models.
4. Use logical inference techniques such as forward/backward chaining and resolution, and apply learning methods like decision trees and reinforcement learning.
5. Understand the architecture and functioning of expert systems, including typical examples like MYCIN and DART.

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 dempstershafer 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", SecondEdition, 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.

Online Learning Resources:

1. https://ai.google/
2. https://swayam.gov.in/nd1_noc19_me71/preview

III Year I Semester	**MICROPROCESSORS & MICROCONTROLLERS (Professional Elective-I)**	**L**	**T**	**P**	**C**
		3	**0**	**0**	**3**

Course Objectives:

1. To introduce the fundamental architectural concepts of microprocessors and microcontrollers.
2. To impart knowledge of the addressing modes and instruction sets of 8086 and 8051.
3. To develop an understanding of assembly language programming concepts.
4. To explain memory and I/O interfacing techniques using 8086 and 8051.
5. To provide an overview of 16-bit and 32-bit microcontrollers, including comparisons with modern architectures like PIC and ARM.

Course Outcomes (COs):

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

1. Explain the internal architecture, operation modes, and interrupt handling of the 8086 microprocessor.
2. Develop assembly language programs for 8086 using various addressing modes and assembler directives.
3. Interface memory and peripheral devices with 8086 using chips like 8255, 8251, 8237A, and 8259, and write relevant I/O programs.
4. Describe the architecture, instruction set, and assembly programming techniques of the 8051 microcontroller.
5. Design and implement real-time interfacing applications using 8051 for devices like LCDs, keypads, sensors, ADC/DACs, and motors, and compare 8051 with PIC and ARM microcontrollers.

UNIT- I:

8086 Architecture: Main features, pin diagram/description, 8086 microprocessor family, internal architecture, bus interfacing UNIT-, execution UNIT-, interrupts and interrupt response, 8086 system timing, minimum mode and maximum mode configuration.

UNIT- II:

8086 Programming: Program development steps, instructions, addressing modes, assembler directives, writing simple programs with an assembler, assembly language program development tools.

UNIT- III:

8086 Interfacing: Semiconductor memories interfacing (RAM, ROM), Intel 8255 programmable peripheral interface, Interfacing switches and LEDS, Interfacing seven segment displays, software and hardware interrupt applications, Intel 8251 USART architecture and interfacing, Intel 8237a DMA controller, stepper motor, A/D and D/A converters, Need for 8259 programmable interrupt controllers.

UNIT- IV:

Microcontroller, Architecture of 8051, Special Function Registers(SFRs), I/O Pins Ports and Circuits, Instruction set, Addressing modes, Assembly language programming.

UNIT- V:

Interfacing Microcontroller, Programming 8051 Timers, Serial Port Programming, Interrupts Programming, LCD & Keyboard Interfacing, ADC, DAC & Sensor Interfacing, External Memory Interface, Stepper Motor and Waveform generation, Comparison of Microprocessor, Microcontroller, PIC and ARM processors

Textbooks:

1. Microprocessors and Interfacing – Programming and Hardware by Douglas V Hall, SSSP Rao, Tata McGraw Hill Education Private Limited, 3rd Edition,1994.
2. K M Bhurchandi, A K Ray, Advanced Microprocessors and Peripherals, 3rd edition, McGraw Hill Education, 2017.
3. Raj Kamal, Microcontrollers: Architecture, Programming, Interfacing and System Design, 2nd edition, Pearson, 2012.

Reference Books:

1. Ramesh S Gaonkar, Microprocessor Architecture Programming and Applications with the 8085, 6th edition, Penram International Publishing, 2013.
2. Kenneth J. Ayala, The 8051 Microcontroller, 3rd edition, Cengage Learning, 2004.

III Year I Semester	DATA WAREHOUSING & DATA MINING (Professional Elective-I)	L	T	P	C
		3	0	0	3

Pre-requisites: Data Structures, Algorithms, Probability & Statistics, Data Base Management Systems

Course Objectives:

The main objectives of this course are to:

1. Introduce the basic concepts and techniques of data warehousing and data mining.
2. Examine the types of data to be mined and apply suitable preprocessing methods on raw data.
3. Enable students to discover interesting patterns using classification, clustering, and association rule mining.
4. Analyze supervised and unsupervised learning models and evaluate the performance of data mining algorithms.
5. Explore data mining applications and technologies with a focus on cloud-based data warehousing and visualization.

Course Outcomes:

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

1. Explain the fundamental concepts of data warehousing, OLAP, and data mining, including data types and similarity measures.
2. Apply data preprocessing techniques such as cleaning, integration, reduction, and transformation to prepare data for mining tasks.
3. Implement and evaluate classification models using decision trees, Bayesian classifiers, and rule-based methods.
4. Generate and analyze association rules using Apriori and FP-Growth algorithms to uncover interesting relationships in large datasets.
5. Perform clustering analysis using algorithms such as K-Means, Hierarchical Clustering, and DBSCAN, and evaluate their strengths and weaknesses.

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 Patten Mining, Technologies, Applications, Major issues, Data Objects & Attribute Types, Basic Statistical Descriptions of Data, Data Visualization, Measuring Data Similarity and Dissimilarity. (Text Book- 1)

UNIT- II:

Data Preprocessing: An Overview, Data Cleaning, Data Integration, Data Reduction, Data Transformation and Data Discretization. (Text Book- 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. (Text Book- 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. (Text Book- 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. (Text Book- 2)

Text Books:

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, Universities Press,2013.
3. (NPTEL course by Prof.PabitraMitra)
   http://onlinecourses.nptel.ac.in/noc17_mg24/preview
4. http://www.saedsayad.com/data_mining_map.htm

III Year I Semester	ADVANCED JAVA LAB	L	T	P	C
		0	0	3	1.5

Course Objectives: the main objectives of the course are

- 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

Lab should cover the following concepts:

- JDBC programming
- J2EE and Web development
- Servlets
- Java Server Pages
- Java Web Frameworks

Sample List of Experiments:

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.

III Year I Semester	Computer Networks Lab	L	T	P	C
		0	0	3	1.5

Course Objectives:

The objectives of this lab course are to:

1. Provide hands-on experience in configuring and analyzing computer networks.
2. Familiarize students with framing methods, error detection/correction techniques, and flow control mechanisms.
3. Enable students to simulate and implement various routing and congestion control algorithms.
4. Introduce network traffic monitoring using tools such as Wireshark.
5. Explore network behavior through simulation tools like NS2/NS3.

Course Outcomes (COs):

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

1. Identify and configure basic network components and apply framing techniques such as character and bit stuffing.
2. Implement and analyze error detection and correction techniques including Hamming Code and various CRC variants.
3. Simulate and evaluate medium access and flow control protocols such as Stop-and-Wait, Go-Back-N, and Selective Repeat.
4. Demonstrate congestion control and routing algorithms such as Leaky Bucket, Dijkstra's, and Distance Vector Routing using simulation or programming.
5. Analyze network traffic using Wireshark and evaluate packet behavior and transmission through NS2/NS3 simulations.

Category 1: Network Basics and Layered Protocols (Foundational)

- Study of Network Devices and LAN Setup
- Data Link Layer Framing Methods – Character Stuffing and Bit Stuffing
- Checksum Generation and Verification

Category 2: Error Detection and Correction

- Hamming Code Implementation
- Cyclic Redundancy Check (CRC) – CRC-12, CRC-16, CRC-CCITT

Category 3: Medium Access & Flow Control

- Stop-and-Wait Protocol Implementation
- Sliding Window Protocol – Go-Back-N
- Sliding Window Protocol – Selective Repeat

Category 4: Congestion and Routing Algorithms

- Leaky Bucket Algorithm for Congestion Control
- Dijkstra's Algorithm for Shortest Path Routing

- Distance Vector Routing Protocol Simulation

Category 5: Traffic Analysis and Tools

- Introduction to Wireshark – Capturing and Analyzing Packets on a Local Network
- Wireshark Packet Analysis – Filtering and Statistics

Category 6: Network Simulation with NS2/NS3

- Simulation of TCP/UDP Behavior and Packet Drop in NS2

Online Resources

- Wireshark Official Labs: https://wiki.wireshark.org/
  Includes tutorials and filters for TCP, HTTP, and DNS.
- NS2/NS3 Documentation and Tutorials:
  - The Network Simulator - ns-2
  - ns-3 Tutorial — Tutorial
- NPTEL Course on Computer Networks (by IIT Professors):
  Computer Networks and Internet Protocol - Course

III Year I Semester	Full Stack Development (Skill Enhancement Course)	L	T	P	C
		0	1	2	2

Course Objectives:

By the end of the course, students will be able to:

- Understand the architecture and components of web development.
- Develop well-structured web pages using HTML and HTML5.
- Create interactive forms and frame-based layouts.
- Integrate multimedia and ensure web accessibility.
- Style web pages effectively using CSS.
- Incorporate JavaScript for interactivity.
- Design and build dynamic user interfaces using React.js.
- Build single-page applications with React Router and manage state using Redux or Context API.
- Integrate frontend applications with RESTful APIs.
- Create backend servers and RESTful APIs using Node.js, Express.js and design MongoDB databases.
- Develop and integrate full-stack MERN applications

UNIT--I :

Web Essentials: Clients, Servers, and Communication. The Internet-Basic Internet Protocols The World Wide Web-HTTP request message-response message-Web Clients Web Servers. Markup Languages: XHTML an Introduction to HTML, History, Versions, Basic, XHTML Syntax and Semantics Some Fundamental HTML Elements-Relative URLs-Lists-tablesFrames-Forms.

UNIT--II:

Style Sheets: CSS-Introduction to Cascading Style Sheets-Features-Core Syntax-Style Sheets and HTML- Style Rule Cascading and Inheritance-Text Properties-Box Model Normal Flow Box Layout beyond the Normal Flow-CSS3.0, Basic Introduction to Java Script.

UNIT--III:

Java Script, React JS Introduction to React, Obstacles and Roadblocks, keeping Up with the Changes, Working with the Files, Pure React, Page Setup, The Virtual DOM, React Elements, ReactDOM, Children, Constructing Elements with Data, React Components, DOM Rendering, Factories.

UNIT- -IV :

Node JS Fundamentals: Event Driven Architecture, Modules, HTTP Module, Express JS: Setting Severs, Routing, Middle Ware, RESTful APIs (GET.POST,PUT,DELETE).

UNIT--V :

MongoDB and MERN Integration, MongoDB Basics: NOSQL Concepts, Collections, Documents, CRUD Operations, Mongoose: Schemas, Models, Querying Data. MERN Integration: Connecting MongoDB, Express JS, React JS, Node JS.

Text Books:

1. Programming the World Wide Web, 7th Edition, Robet W Sebesta, Pearson, 2013.
2. Web Programming with HTML5, CSS and JavaScript, John Dean, Jones & Bartlett Learning, 2019 (Chapters 1-11)
3. JavaScript: The Good Parts, Douglas Crockford,1st Edition (2008), O'Reilly Media.
4. Web Development with Node and Express, Ethan Brown, 2nd Edition (2019), O'Reilly.

Reference Text Books:

1. Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node, Vasan Subramanian, 2nd edition, APress, O'Reilly.
2. Learning React: Modern Patterns for Developing React Apps, Alex Banks, Eve Porcello, 2nd Edition (2020), O'Reilly.
3. MongoDB: The Definitive Guide, Shannon Bradshaw, Kristina Chodorow, 3rd Edition (2019), 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/jquery/default.asp
5. https://www.w3schools.com/react/default.asp
6. https://www.w3schools.com/nodejs/default.asp
7. https://apidog.com/blog/how-to-create-a-rest-api-with-node-js-and-express/?utm_source=google_dsa&utm_medium=s&utm_campaign=22062217351&utm_content=169453484141&utm_term=&gad_source=5&gad_campaignid=22062217351&gclid=EAIaIQobChMI0O760__1jQMVd6RmAh0xRwTxEAAYASAAEgLscPD_BwE

III Year I Semester	USER INTERFACE DESIGN USING FLUTTER	L	T	P	C
		0	0	2	1

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 1st Edition, Apres
3. Richard Rose, Flutter & Dart Cookbook, Developing Full stack Applications for the Cloud, Oreilly.

III Year II Semester	ADVANCED DATA STRUCTURES	L	T	P	C
		3	0	0	3

Course Objectives

1. Understand fundamental and advanced data structures, their operations, and complexity analysis.
2. Analyze algorithms using time and space complexity, recurrence relations, and amortized techniques.
3. Apply advanced data structures like balanced trees, heaps, and disjoint sets to real-world problems.
4. Explore efficient indexing and searching using hashing techniques and trie-based structures.
5. Utilize graph data structures and algorithms for solving computational problems in various domains.

Course Outcomes

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

1. Demonstrate proficiency in using linear and non-linear data structures for problem-solving.
2. Analyze the performance of algorithms using complexity measures, recurrence relations, and amortized analysis.
3. Implement and apply balanced tree structures and advanced heap data structures in algorithm design.
4. Design efficient indexing and search solutions using tries, hashing, and suffix structures.
5. Solve real-world problems using appropriate graph representations and algorithms including MST and shortest path algorithms.

UNIT- I :

Introduction and Algorithm Analysis-Review of basic data structures – arrays, stacks, queues, linked lists, trees, and their operations. Time and space complexity analysis. Recurrence relations and solving them using the Master Theorem. Amortized analysis – aggregate, accounting, and potential methods.

UNIT- II :

Balanced Trees and Indexing Structures-Advanced tree structures – AVL Trees, Red-Black Trees, Splay Trees. B-Trees and B+ Trees for indexing in databases and file systems. Segment Trees and Fenwick Trees (Binary Indexed Trees) for efficient range query processing.

UNIT- III:

Priority Queues and Disjoint Sets- Binary Heaps, Binomial Heaps, Fibonacci Heaps – operations and efficiency. Applications in priority queues and Dijkstra's algorithm. Disjoint Set ADT – union by rank and path compression. Applications in Kruskal's algorithm and network connectivity problems.

UNIT - IV:

Hashing and Tries-Advanced hashing techniques – open addressing, double hashing, cuckoo hashing. Hash functions and collision resolution. Tries, compressed tries, suffix tries, and ternary search trees. Applications in text search and dictionary implementations. Introduction to suffix arrays and Bloom filters.

UNIT- V:

Graph Data Structures and Applications-Representation of graphs – adjacency matrix, list, and edge list. Traversal algorithms – BFS, DFS, and their applications. Advanced concepts – shortest path (Dijkstra, Bellman-Ford), MST (Prim, Kruskal), and Union-Find. Real-world applications in networks, compilers, and social media analysis.

Textbooks

1. Mark Allen Weiss, *Data Structures and Algorithm Analysis in C++*, Pearson, 4th Edition
2. Thomas H. Cormen et al., *Introduction to Algorithms*, MIT Press, 3rd Edition

Reference Books

- Alfred Aho, Jeffrey Ullman, *Data Structures and Algorithms*
- S. Sahni and E. Horowitz, *Fundamentals of Data Structures*
- Steven S. Skiena, *The Algorithm Design Manual*

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

Course Objectives

The primary objectives of this course are to:

1. Understand the **fundamental principles** and **mathematical foundations** of cryptography.
2. Explore **symmetric** and **asymmetric encryption** techniques and their cryptographic algorithms.
3. Study the design and applications of **cryptographic hash functions**, **digital signatures**, and **key management protocols**.
4. Learn about **authentication protocols** and **secure communication standards** like Kerberos, IPsec, SSL, and TLS.
5. Examine real-world applications of cryptography in **network security**, **system security**, and **secure messaging**.

Course Outcomes

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

1. Explain the goals of cryptography and classify various attacks, services, and mechanisms.
2. Analyze and apply symmetric encryption techniques such as DES and AES along with the supporting algebraic structures.
3. Demonstrate the working of asymmetric encryption algorithms like RSA, ElGamal, and ECC, and understand the number-theoretic concepts underlying them.
4. Assess the integrity and authentication mechanisms through hash functions, digital signatures, and key management protocols.
5. Apply cryptographic concepts in network security protocols at different layers , and explain modern threats and system-level security techniques like firewalls andIDS.

UNIT- I:

Basic Principles: Security Goals, Cryptographic Attacks, Services and Mechanisms, Mathematics of Cryptography- integer arithmetic, modular arithmetic, matrices, linear conguence.

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

Reference Books:

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

III Year II Semester	MACHINE LEARNING	L	T	P	C
		3	0	0	3

Course Objectives:

The objectives of the course are to:

1. Define the foundational concepts of **machine learning**, including supervised, unsupervised, and reinforcement learning paradigms.
2. Understand and apply **distance-based algorithms** like **K-Nearest Neighbors (KNN)** for classification and regression.
3. Explore decision tree-based models including **Random Forests** and **Naïve Bayes Classifier**, and analyze their behavior.
4. Gain insight into **linear models** such as **Perceptron**, **Logistic Regression**, and **Support Vector Machines**, along with neural network basics.
5. Implement various **clustering techniques**, including **K-means**, **Fuzzy C-means**, **Rough Clustering**, and **EM-based Clustering**.

Course Outcomes:

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

1. Explain the key concepts, types, and stages of machine learning and their real-world applications.
2. Apply and evaluate proximity-based algorithms like **KNN** for classification and regression problems.
3. Build and analyze decision trees, **Random Forests**, and **Naïve Bayes classifiers** for classification tasks.
4. Implement linear classifiers like **Perceptron**, **SVM**, and **Logistic Regression** and understand neural network training using backpropagation.
5. Demonstrate the ability to apply unsupervised learning algorithms for clustering tasks using **K-Means**, **Fuzzy C-Means**, **Spectral Clustering**, and others.

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, 7th Edition, 2019.

III Year II Semester	SOFTWARE TESTING METHODOLOGIES (Professional Elective-II)	L	T	P	C
		3	0	0	3

Course Objectives

The main objectives of this course are to:

1. Understand the **fundamental concepts** of software testing and its role in software quality assurance.
2. Introduce different **testing models**, types of bugs, and a taxonomy to classify bugs.
3. Learn and apply **flow graph-based testing**, including **path testing** and **data flow testing**.
4. Explore **domain testing**, **logic-based testing**, and **state transition testing** techniques.
5. Study advanced testing topics like **graph matrices**, **regular expressions**, and their application in tool-based testing.
6. Provide hands-on exposure to **automated software testing tools** such as **JMeter**, **Selenium**, **SoapUI**, or **Catalon**.

Course Outcomes:

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

1. Apply **path testing techniques** using flow graphs and predicates to design test cases.
2. Perform **transaction flow testing**, **data flow testing**, and **domain testing** to assess program correctness and detect anomalies.
3. Analyze software behavior using **path expressions**, **logic-based testing**, and **decision tables**.
4. Apply **state-based and transition testing** for modeling and verifying software systems.
5. Use **graph matrices**, **node reduction algorithms**, and **automated tools** for effective test design and execution.

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 - Baris Beizer, 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.

III Year II Semester	AUGMENTED REALITY & VIRTUAL REALITY (Professional Elective-II)	L	T	P	C
		3	0	0	3

Course Objectives:

The objectives of this course are to:

1. Provide a strong **foundation in the concepts, history, and components** of Augmented Reality (AR).
2. Introduce students to the **fundamentals of Virtual Reality (VR)** including its technical, perceptual, and physiological aspects.
3. Explain the **hardware and software components** used in AR/VR, such as displays, tracking systems, and computer vision techniques.
4. Describe the **human sensory and perceptual systems**, and how they influence AR/VR system design.
5. Expose students to **interaction techniques** and **audio-visual rendering** used in immersive systems.

Course Outcomes:

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

1. Explain the fundamental concepts, history, components, and tracking technologies involved in Augmented Reality systems.
2. Apply computer vision techniques for AR, understand interaction modalities, and describe AR software architectures.
3. Understand the historical context and mathematical modeling of Virtual Reality including viewing geometry, optics, and light behavior.
4. Analyze human visual physiology and visual perception for realistic rendering and immersive experiences in VR.
5. Apply concepts of motion, interaction, and auditory rendering to enhance immersion in virtual environments.

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 Development‖, 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. Sanni Siltanen- 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

III Year II Semester	DEVOPS (Professional Elective-II)	L	T	P	C
		3	0	0	3

Course Objectives:

The objectives of this course are to:

1. Understand the principles of **DevOps**, its integration with Agile and SDLC, and its significance in modern software delivery.
2. Gain hands-on knowledge of **source code management** with Git and **UNIT-testing tools** like JUNIT-, NUNIT-, and SonarQube.
3. Learn the automation of **builds** and **Continuous Integration (CI)** using Jenkins and related tools.
4. Explore **Continuous Delivery (CD)** processes, **containerization using Docker**, and testing automation using tools like Selenium.
5. Implement **Configuration Management** with tools like **Ansible**, and understand **container orchestration** with **Kubernetes/OpenShift** along with Puppet and Chef.

Course Outcomes:

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

1. Explain the core concepts of DevOps, its lifecycle, architecture, and integration with Agile methodologies and software development workflows.
2. Use Git for version control and collaborate using Git commands, branching, and staging; apply UNIT- testing and analyze code quality using SonarQube.
3. Implement build automation and Continuous Integration using Jenkins pipelines, master-slave architecture, and manage Jenkins jobs and builds.
4. Demonstrate continuous delivery and deployment using Docker containers, and perform automated testing using tools like Selenium.
5. Apply configuration management using Ansible, and orchestrate containerized applications using Kubernetes/OpenShift; understand Puppet and Chef basics.

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 Cl 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, 1st Edition Mihails Konoplows, 2015.
2. Alisson Machado de Menezes., Hands-on DevOps with Linux,1st 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, 1st Edition, Packt Publishing, 2016.

4. Joakim Verona. Practical Devops, Ingram short title; 2nd edition (2018). ISBN10: 1788392574
5. Deepak Gaikwad, Viral Thakkar. DevOps Tools from Practitioner's Viewpoint. Wiley publications. ISBN: 9788126579952

III Year II Semester	GENERATIVE AI (Professional Elective-II)	L	T	P	C
		3	0	0	3

Course Objectives:

The main objectives of the course are to:

1. Provide a **comprehensive understanding** of generative AI, its models, use cases, and ethical implications.
2. Explore **generative models for text** including transformers, attention mechanisms, GPT/BERT, and prompt engineering techniques.
3. Introduce **image generation** using GANs, VAEs, diffusion models, and transformer-based visual generation architectures.
4. Study **creative applications** of Gen AI in generating art, music, and autonomous agents using RNNs, GANs, and reinforcement learning.
5. Familiarize students with **open-source frameworks**, model fine-tuning, LLM programming, and deployment tools like Hugging Face and LangChain.

Course Outcomes:

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

1. Explain the fundamentals of generative AI, its historical development, key model types (GANs, VAEs, autoregressive, diffusion), and ethical implications in AI.
2. Analyze and apply generative text models such as GPT and BERT; demonstrate prompt engineering, transformer architecture, and tackle LLM-specific challenges.
3. Illustrate image generation using GANs, VAEs, Stable Diffusion, and transformer-based vision models like CLIP, DALL·E, and GPT-4V.
4. Explore generative models for creative domains including painting, music, and games using neural style transfer, MuseGAN, and reinforcement learning.
5. Utilize and fine-tune open-source generative models (like LLaMA, GPT4All); implement solutions using frameworks like LangChain, Hugging Face, and deploy GenAI applications.

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 of 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.

III Year II Semester	INTELLIGENT AI AGENTS (Professional Elective-II)	L	T	P	C
		3	0	0	3

Course Objectives:

The primary objectives of this course are to:

1. Introduce the concept, characteristics, and classifications of intelligent agents and their interaction with environments.
2. Explore different **agent architectures** including reflex, goal-based, utility-based, and hybrid designs.
3. Understand foundational concepts of **knowledge representation and planning** using logic, ontologies, and classical planning methods.
4. Study various learning paradigms in agents, including **reinforcement learning**, Q-learning, and deep learning-based models.
5. Provide exposure to **real-world applications** of intelligent agents and multi-agent systems in robotics, games, IoT, and conversational AI.

Course Outcomes :

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

1. Explain the characteristics, types, and models of intelligent agents and differentiate between various agent environments.
2. Analyze and compare various agent architectures including reflex, goal-based, utility-based, and hybrid models.
3. Apply knowledge representation techniques and formulate planning problems using classical and probabilistic models.
4. Implement learning methods for agents using supervised, unsupervised, and reinforcement learning including Q-learning and policy gradients.
5. Demonstrate the use of intelligent agents in applications such as games, robotics, and IoT using simulation tools and assess ethical considerations.

UNIT-- I:

Introduction to Intelligent Agents-Definition and characteristics of intelligent agents, Types of agents: reactive, deliberative, hybrid, learning, PEAS (Performance measure, Environment, Actuators, Sensors) model, Agent environments: deterministic vs. stochastic, episodic vs. sequential, etc., Multi-agent systems: basic concepts

UNIT-- II:

Agent Architectures-Simple reflex agents, Model-based reflex agents, Goal-based and utility-based agents, Layered and hybrid architectures (Subsumption, InteRRaP, BDI), Behavior-based and reactive planning.

UNIT-- III:

Knowledge Representation and Planning-Propositional and predicate logic, Semantic networks and ontologies, Planning problem formulation, Classical planning: STRIPS, Graphplan, Partial-order planning, Planning under uncertainty (MDP, POMDP basics)

UNIT- IV:

Learning in Intelligent Agents-Supervised, unsupervised, and reinforcement learning, Q-learning and deep Q-networks (DQN),Policy gradient and actor-critic methods, Learning agent models (exploration vs. exploitation),Adaptation in dynamic environments

UNIT- V:

Applications and Tools-Game-playing agents, Agents in robotics and IoT, Chatbots and conversational agents, Simulation platforms: NetLogo, MESA, OpenAI Gym, UNIT-y ML-Agents, Ethical and societal implications of autonomous agents

Textbooks:

1. **Stuart Russell & Peter Norvig**, *Artificial Intelligence: A Modern Approach*, Pearson, 4th Edition, 2020.
2. **Michael Wooldridge**, *An Introduction to MultiAgent Systems*, Wiley, 2nd Edition, 2009.

Reference Books & Resources:

- **Nils J. Nilsson**, *The Quest for Artificial Intelligence*, Cambridge University Press.
- **Gerhard Weiss**, *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*, MIT Press.
- Research papers from *AAAI*, *IJCAI*, *AAMAS*, *NeurIPS*.
- Online courses: Coursera (Reinforcement Learning, Multi-Agent Systems), edX, Stanford CS221.

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

Course Objectives:

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

1. Understand the principles, practices, and evolution of software project management, including conventional and modern approaches.
2. Describe and analyze various life cycle phases and the associated artifacts produced during a software project.
3. Apply planning, estimation, and monitoring techniques to manage workflows, milestones, and iterations in software projects.
4. Evaluate different organizational structures, automation tools, and performance metrics for effective project control and team management.
5. Analyze and apply Agile and DevOps methodologies in real-world project environments to enhance flexibility, collaboration, and delivery speed.

Course Outcomes:

After completing this course, students will be able to:

1. Describe traditional and modern approaches in software project management, and analyze how improvements in software economics can be achieved.
2. Explain various life cycle phases of software projects and categorize the associated management and engineering artifacts.
3. Apply iterative planning principles and estimation techniques to define and manage project workflows, milestones, and schedules.
4. Evaluate different organizational structures, understand the role of process automation, and interpret project control metrics for better decision-making.
5. Demonstrate the application of Agile and DevOps practices in project environments, including deployment pipelines and team collaboration frameworks.

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

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

Course Objectives:

From this course, the student will learn:

1. How to architect sensor networks tailored to specific application scenarios.
2. Methods to devise suitable data dissemination protocols and efficiently model communication link costs.
3. The fundamental concepts of wireless sensor networks (WSNs) including architecture, protocols, and operational mechanisms at different layers.
4. How to evaluate the performance of sensor networks, identify performance bottlenecks, and propose solutions.
5. The security challenges associated with both Ad Hoc and wireless sensor networks and the mechanisms to mitigate them.

Course Outcomes:

After completing this course, the student will be able to:

1. Explain the characteristics, applications, and design challenges of Mobile Ad Hoc Networks (MANETs) and describe MAC layer protocols suited for such environments.
2. Classify and analyze routing and transport layer protocols in ad hoc networks and evaluate their suitability under varying network conditions.
3. Describe common security threats in ad hoc networks and apply secure routing protocols, intrusion detection mechanisms, and key management techniques.
4. Understand the structure and functioning of wireless sensor networks including sensor node design, communication range, clustering, and data retrieval mechanisms.
5. Apply knowledge of security protocols, sensor node operating systems, and simulation tools like NS-2 and TOSSIM for evaluating and prototyping WSN applications.

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

Reference Books:

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

5. Wireless Sensor Networks – Principles and Practice, 1st edition, Fei Hu, Xiaojun Cao, An Auerbach book, CRC Press, Taylor & Francis Group, 2010

III Year II Semester	NATURAL LANGUAGE PROCESSING (Professional Elective-III)	L	T	P	C
		3	0	0	3

Course Objectives:

The course is designed to:

1. Provide foundational knowledge on Natural Language Processing and its real-world challenges.
2. Introduce language models, morphological analysis, and spelling correction techniques.
3. Equip students with methods for word-level analysis, including N-gram models and part-of-speech tagging.
4. Explore syntactic and semantic processing techniques, including parsing and meaning representation.
5. Examine discourse-level phenomena and introduce lexical resources and tools for advanced language understanding.

Course Outcomes:

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

1. Understand and apply core concepts of NLP including tokenization, morphology, and spelling correction.
2. Develop and evaluate statistical language models and use them in various NLP tasks.
3. Implement part-of-speech tagging using rule-based and probabilistic models.
4. Analyze and construct syntactic structures using context-free and dependency grammars.
5. Represent meaning through semantic frameworks and resolve word sense ambiguities.

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, 2nd Edition, Daniel Jurafsky, James H. Martin -Pearson Publication,2024.
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, 1st Edition, Breck Baldwin, Atlantic Publisher, 2015.
2. Natural Language Processing with Java, 2nd Edition, Richard M Reese, OReilly Media,2015.
3. Handbook of Natural Language Processing, Second, NitinIndurkhya and Fred J. Damerau, Chapman and Hall/CRC Press, 2010.Edition
4. Natural Language Processing and Information Retrieval, 3rd Edition, TanveerSiddiqui, U.S. Tiwary, Oxford University Press,2008.

III Year II Semester	DISTRIBUTED OPERATING SYSTEM (Professional Elective-III)	L	T	P	C
		3	0	0	3

Course Objectives:

This course aims to:

1. Understand the fundamental principles and evolution of distributed computing systems and operating systems.
2. Learn about message passing mechanisms and inter-process communication in distributed environments.
3. Explore remote procedure calls and their implementation in distributed client-server models.
4. Analyze the design and implementation of Distributed Shared Memory (DSM) systems and synchronization mechanisms.
5. Examine resource management strategies and distributed file system design including fault tolerance and replication.

Course Outcomes :

After completing this course, the student will be able to:

1. Explain the architecture, models, and issues in distributed computing and operating systems.
2. Apply message passing techniques and remote procedure call mechanisms in distributed systems.
3. Analyze distributed shared memory structures, consistency models, and synchronization methods.
4. Evaluate resource management strategies including scheduling, process migration, and threading.
5. Demonstrate understanding of distributed file systems with respect to sharing, replication, caching, and fault tolerance.

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 Learning, Latest Reprint (2023 or most recent), ISBN: 978-8120313804**.
2. **George Coulouris, Jean Dollimore, Tim Kindberg, Gordon Blair**, *Distributed Systems: Concepts and Design*, **Pearson Education, 5th Edition, Latest Indian Reprint (2020 or newer), ISBN: 978-9353437350**.

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, 2008
3. SUNIT-aMahajan, Seema Shan, " Distributed Computing", Oxford University Press,2015

III Year II Semester	ADVANCED DATA STRUCTURES LAB	L	T	P	C
		0	0	3	1.5

Course Objectives:

- To gain practical experience in implementing and debugging advanced data structures.
- To develop skills in analyzing the time and space complexity of different implementations.
- To strengthen problem-solving abilities through competitive coding techniques.

List of Suggested Lab Experiments

1. Implementation of AVL Tree with insertion and deletion
2. Implementation of Red-Black Tree
3. Construction and operations on B-Trees and B+ Trees
4. Segment Tree and Fenwick Tree implementation for range queries
5. Binary Heap and Priority Queue operations
6. Union-Find operations with path compression
7. Open Addressing and Cuckoo Hashing
8. Trie and Suffix Trie implementation
9. Dijkstra's and Bellman-Ford shortest path algorithms
10. Kruskal's and Prim's algorithm for Minimum Spanning Tree
11. BFS and DFS traversal of graphs
12. Project: Design and implement a mini-application (e.g., autocomplete search, dynamic connectivity checker)

Note: Lab may be implemented in C/Java/Python as per Teacher preference.

III Year II Semester	MACHINE LEARNING LAB	L	T	P	C
		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

III Year II Semester	SOFT SKILLS (Skill Enhancement Course)	L	T	P	C
		0	1	2	2

Course Objectives:

This course aims to:

1. Develop students' analytical thinking, listening, and verbal/non-verbal communication skills.
2. Equip students with essential self-management strategies including stress and time management.
3. Improve grammatical and writing skills for professional and academic settings.
4. Prepare students for employment through training in group discussions, resume writing, and interview techniques.
5. Enhance interpersonal effectiveness and relationship management skills.

Course Outcomes (COs):

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

1. Demonstrate improved listening, analytical thinking, and communication skills through verbal and non-verbal expressions.
2. Apply self-management techniques such as stress, anger, and time management to handle personal and professional challenges effectively.
3. Use standard English grammar, writing, and note-making skills to communicate clearly in written formats.
4. Prepare job-oriented documents and perform confidently in group discussions and interviews.
5. Establish and maintain positive interpersonal relationships by understanding different styles and factors influencing them.

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

III Year II Semester	TECHNICAL PAPER WRITING & IPR	L	T	P	C
		2	0	0	-

Course Objectives:

1. To introduce the fundamentals of technical report writing, including sentence formation and tense usage.
2. To help students understand the principles of formatting and structuring technical documents.
3. To develop skills in proofreading, summarizing, and proposal writing.
4. To enable students to effectively use word processors for professional document preparation and presentation.
5. To create awareness about Intellectual Property Rights (IPR) and the process of patenting and innovation.

Course Outcomes :

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

1. Understand the fundamentals of technical writing, sentence formation, tense usage, and report structure with clarity and precision.
2. Draft and refine technical documents using appropriate formatting, visuals, and plain English principles to enhance clarity and readability.
3. Apply proofreading techniques and summarization skills while preparing and presenting final technical reports and proposals effectively.
4. Utilize advanced word processing tools for organizing, editing, and securing technical documents professionally.
5. Understand the basic concepts of Intellectual Property Rights (IPR), including patents, copyrights, and international cooperation mechanisms.

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", 1st 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", 2nd 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/

IV Year I Semester	INTERNET OF THINGS	L	T	P	C
		3	0	0	3

Course Objectives:

1. To understand the evolution, components, and enabling technologies of IoT.
2. To study sensing, actuation, and processing mechanisms in IoT systems.
3. To explore various IoT connectivity and communication protocols.
4. To examine interoperability standards and fog computing in IoT.
5. To analyze emerging IoT paradigms, challenges, and real-world applications.

Course Outcomes:

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

1. Explain the foundational concepts and technological evolution of IoT.
2. Identify and describe sensors, actuators, and processing strategies used in IoT devices.
3. Select suitable connectivity and communication technologies for specific IoT applications.
4. Apply interoperability frameworks and fog computing concepts to design IoT systems.
5. Analyze current trends, challenges, and apply IoT solutions to real-world domains like agriculture and transportation.

UNIT- I:

Predecessors of IoT: Introduction, Wireless Sensor Networks, Machine-to-Machine Communications, Cyber Physical Systems

Emergence of IoT: Introduction, Evolution of IoT, Enabling IoT and the Complex Interdependence of Technologies, IoT Networking Components, Addressing Strategies in IoT

UNIT- II:

IoT Sensing and Actuation: Introduction, Sensors, Sensor Characteristics, Sensorial Deviations, Sensing Types, Sensing Considerations, Actuators, Actuator Types, Actuator Characteristics

IoT Processing Topologies and Types: Data Format, Importance of Processing in IoT, Processing Topologies, IoT Device Design and Selection Considerations, Processing Offloading.

UNIT- III:

IoT Connectivity Technologies: Introduction, IEEE 802.15.4, Zigbee, Thread, ISA100.11A, WirelessHART, RFID, NFC,DASH7, Z-Wave, Weightless, Sigfox, LoRa, NB-IT, Wi-Fi, Bluetooth

IoT Communication Technologies: Introduction, Infrastructure Protocols, Discovery Protocols, Data Protocols, Identification Protocols, Device Management, Semantic Protocols.

UNIT- IV:

IoT Interoperability: Introduction, Standards, Frameworks

Fog Computing and Its Applications: Introduction, View of Fog Computing Architecture, Fog Computing in IoT, Selected Applications of Fog Computing

UNIT- V:

Paradigms, Challenges, and the Future: Introduction, Evolution of New IoT Paradigms, Challenges Associated with IoT, Emerging Pillars of IoT

IoT Case Studies: Agricultural IoT, Vehicular IoT

Text Books:

1. Introduction to IoT, Sudip Misra, Anandarup Mukhaerjee, Arjit Roy, Cambridge University Press, 2021
2. Internet of Things: Architecture, Design Principles and Applications, Rajkamal, McGraw Hill Higher Education

Reference Books:

1. Fog and Edge Computing: Principles and Paradigms, Rajkumar Buyya (Editor), Satish narayana Srirama (Editor) , ISBN: 978-1-119-52498-4, January 2019
2. Getting Started with the Internet of Things, CunoPfister , Oreilly

IV Year I Semester	HUMAN RESOURCES & PROJECT MANAGEMENT	L	T	P	C
		3	0	0	3

Course Objectives:

The main objectives of the course are to

1. Provide knowledge about HR planning, recruitment, selection, and job design.
2. Develop skills in managing HR functions such as performance appraisal, compensation, and employee relations.
3. Emphasize the importance of ethical practices and HR audits in maintaining organizational health.
4. Understand the HRD framework and its impact on organizational success.
5. **Improve group interaction and team dynamics** for better collaboration and performance.
6. Understand the Fundamentals of Project Management and Project Networks
7. Implement appropriate management strategies tailored to specific challenges in different project types.

Course Outcomes:

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

1. **Explain** the principles and functions of Human Resource Management, including HR planning, recruitment, selection, and job design.
2. **Demonstrate** effective HR practices such as training & development, performance appraisal, compensation, and employee relations in organizational contexts.
3. **Apply** ethical standards, HR audit models, and Human Resource Development (HRD) frameworks to assess and improve organizational performance.
4. **Understand and utilize** project management fundamentals including project life cycle, planning, monitoring, and control for various types of projects.
5. **Evaluate** project implementation strategies, human factors, and review mechanisms for successful execution and closure of projects.

UNIT –I:

HRM: Nature, Scope, Concept of HRM, Functions of HRM, Role of HR manager, emerging trends in HRM, E-HRM, HR audit models, ethical aspects of HRM. HR Planning, Demand and Supply forecasting of HR, Job Design, Recruitment, Sources of recruitment, Selection- Selection Procedure.

UNIT –II:

HRD, HR accounting, Models, Concept of Training and Development, Methods of Training. Performance Appraisal: Importance Methods of performance appraisal,

Career Development and Counseling, group interaction.

UNIT –III:

Basics of Project Management, Concept, resource management, Project environment, Types of Projects, project networks-DPR, Project life cycle, Project proposals, Monitoring project progress, Project appraisal and Project selection, 80-20 rules, production technology, communication matrix

UNIT-IV:

Identify various project types and their unique management challenges and apply appropriate management strategies for each. Project Implementation and Review: Forms of project organization, project planning, project control, human aspects of project management, prerequisites for successful project implementation, project review, performance evaluation, abandonment analysis

UNIT-V:

Project Implementation and Review: Forms of project organization, project planning, project control, human aspects of project management, prerequisites for successful project implementation, project review, performance evaluation, abandonment analysis

Text Books:

1. Robert L. Mathis, John H. Jackson, Manas Ranjan Tripathy, Human Resource Management, Cengage Learning 2016.
2. Sharon Pande and Swapnalekha Basak, Human Resource Management, Text and Cases, Vikas Publishing, 2e, 2016.
3. Stewart R. Clegg, Torgeir Skyttermoen, Anne Live Vaagaasar, Project Management, Sage Publications, 1e, 2021.
4. K. Nagarajan, Project Management, New Age International Publishers, 8e, 2017.

Reference Books :

1. Subba Rao P, “Personnel and Human Resource Management-Text and Cases”, Himalaya Publications, Mumbai, 2013.
2. K Aswathappa, “Human Resource and Personnel Management”, Tata McGraw Hill, New Delhi, 2013.
3. Prasanna Chandra, “Projects, Planning, Analysis, Selection, Financing, Implementation and Review”, Tata McGraw Hill Company Pvt. Ltd., New Delhi, 1998.
4. Vasanth Desai, “Project Management”, 4th edition, Himalaya Publications, 2018.
5. Lalitha Balakrishnan, Gowri, “Project Management”, Himalaya publishing house, New Delhi, 2022.

IV Year I Semester	SOFTWARE ARCHITECTURE & DESIGN PATTERNS (Professional Elective-IV)	L	T	P	C
		3	0	0	3

Course Objectives:

The main objectives of the course are to:

1. Introduce the concept, types, and organization of **design patterns** in software development.
2. Explain the **object-oriented development paradigm**, its concepts, benefits, and drawbacks.
3. Provide insight into the **analysis and design phases** of object-oriented software development.
4. Explore and apply various **structural and architectural design patterns** including MVC.
5. Understand the design and implementation of **distributed object-oriented systems** and web services.

Course Outcomes:

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

1. **Define and classify** design patterns and explain their role in solving recurring design problems.
2. **Analyze and model** system requirements using object-oriented principles and domain knowledge.
3. **Apply** structural design patterns such as Adapter, Facade, and Proxy in software solutions.
4. **Design interactive systems** using the MVC architectural pattern and implement its components effectively.
5. **Develop distributed applications** using Java RMI, SOAP/RESTful web services, and integrate them using an Enterprise Service Bus.

UNIT- I

Introduction: What is a design pattern? Describing design patterns, the catalog of design pattern, organizing the catalog, how design patterns solve design problems, how to select a design pattern, how to use a design pattern What is object oriented development? key concepts of object oriented design other related concepts, benefits and drawbacks of the paradigm

UNIT – II

Analysis a System: Overview of the analysis phase, stage 1 gathering the requirements functional requirements specification, defining conceptual classes and relationships,

using the knowledge of the domain Design and Implementation, discussions and further reading

UNIT – III

Design Pattern Catalog: Structural patterns, Adapter, bridge, composite, decorator, facade, flyweight, proxy.

UNIT – IV

Interactive systems and the MVC architecture: Introduction The MVC architectural pattern, analyzing a simple drawing program designing the system, designing of the subsystems, getting into implementation, implementing undo operation drawing incomplete items, adding a new feature pattern based solutions

UNIT – V

Designing with Distributed Objects: Client server system, java remote method invocation, implementing an object oriented system on the web, Web services (SOAP, Restful), Enterprise Service Bus

Text Books:

1. Object oriented analysis, design and implementation, brahma dathan, sarnath rammath , universities press,2013
2. Design patterns, Erich Gamma, Richard helan , Ralph johman , john vlissides, PEARSON Publication,2013

Reference Books:

1. Frank Bachmann, Regine Meunier , Hans Rohnert “Pattern Oriented Software Architecture” Volume 1, 1996.
2. William J Brown et al., "Anti Patterns: Refactoring Software, Architectures and Projects in Crisis", John Wiley, 1998

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

Course Objectives:

The course is designed to:

1. Provide foundational understanding of **machine learning evolution** and its transition into deep learning.
2. Introduce the principles of **artificial neural networks** and the training of deep networks.
3. Enable practical implementation of **deep learning models** using popular frameworks like Keras, TensorFlow, and PyTorch.
4. Explore advanced neural network architectures including **CNNs, RNNs, LSTMs, and Attention mechanisms**.
5. Examine real-world applications and emerging research areas such as **GANs, NLP, deep reinforcement learning**, and **autoencoders**.

Course Outcomes:

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

1. **Explain** the historical development and core principles of machine learning and deep learning.
2. **Design and train** artificial neural networks using suitable architectures for classification tasks.
3. **Implement and evaluate** deep learning models using Keras, TensorFlow, and PyTorch for various use cases.
4. **Apply** convolutional and recurrent neural networks to vision and sequence modeling problems.
5. **Analyze and experiment** with advanced deep learning applications like GANs, reinforcement learning, and generative models.

UNIT- I:

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

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

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

UNIT- IV:

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

UNIT- V:

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

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

Text Books:

1. Deep Learning- Ian Goodfellow, Yoshua Bengio and Aaron Courvile, MIT Press, 2016
2. Deep Learning with Python - Francois Chollet, Released December 2017, Publisher(s): Manning Publications, ISBN: 9781617294433
3. Deep Learning Illustrated: A Visual, Interactive Guide to Artificial Intelligence - Jon Krohn, Grant Beyleveld, Aglaé Bassens, Released September 2019, Publisher(s): Addison-Wesley Professional, ISBN: 9780135116821
4. Deep Learning from Scratch - Seth Weidman, Released September 2019, Publisher(s): O'Reilly Media, Inc., ISBN: 9781492041412

Reference Books:

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

Web Link:

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

IV Year I Semester	COMPUTER VISION (Professional Elective-IV)	L	T	P	C
		3	0	0	3

Course Objectives:

The course is designed to:

1. Introduce the fundamentals of **camera models**, **radiometry**, and **color representation** in imaging systems.
2. Develop an understanding of **image processing techniques** including filtering, edge detection, and texture analysis.
3. Explain the geometry of **multiple views**, **stereopsis**, and **image segmentation techniques**.
4. Familiarize students with **model fitting**, **tracking algorithms**, and **probabilistic methods** for inference.
5. Apply geometric camera models, calibration techniques, and model-based vision to **real-world vision applications**.

Course Outcomes:

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

1. **Describe** radiometry principles, light interaction with surfaces, and color models used in image formation.
2. **Apply** linear filters, Fourier transforms, edge detection, and texture analysis for image interpretation.
3. **Analyze** stereo vision systems and implement segmentation using clustering and graph-based methods.
4. **Use** probabilistic approaches and the EM algorithm for tracking and robust model fitting in dynamic systems.
5. **Calibrate** cameras using geometric models and solve vision problems in domains such as robotics and medical imaging.

UNIT –I:

Cameras: Pinhole Cameras Radiometry – Measuring Light: Light in Space, Light Surfaces, Important Special Cases Sources, Shadows, And Shading: Qualitative Radiometry, Sources and Their Effects, Local Shading Models, Application: Photometric Stereo, Interreflections: Global Shading Models Color: The Physics of Color, Human Color Perception, Representing Color, A Model for Image Color, Surface Color from Image Color.

UNIT-II:

Linear Filters: Linear Filters and Convolution, Shift Invariant Linear Systems, Spatial Frequency and Fourier Transforms, Sampling and Aliasing, Filters as Templates, Edge **Detection**:Noise, Estimating Derivatives, Detecting Edges Texture0:Representing

Texture, Analysis (and Synthesis) Using Oriented Pyramids, Application: Synthesis by Sampling Local Models, Shape from Texture.

UNIT-III:

The Geometry of Multiple Views: Two Views Stereopsis: Reconstruction, Human Stereopsis, Binocular Fusion, Using More Cameras Segmentation by Clustering: What Is Segmentation?

Human Vision: Grouping and Getstalt, Applications: Shot Boundary Detection and Background Subtraction, Image Segmentation by Clustering Pixels, Segmentation by Graph-Theoretic Clustering,

UNIT-IV:

Segmentation by Fitting a Model: The Hough Transform, Fitting Lines, Fitting Curves, Fitting as a Probabilistic Inference Problem, Robustness Segmentation and Fitting Using **Probabilistic Methods:** Missing Data Problems, Fitting, and Segmentation, The EM Algorithm in Practice,

Tracking With Linear Dynamic Models: Tracking as an Abstract Inference Problem, Linear Dynamic Models, Kalman Filtering, Data Association, Applications and Examples

UNIT- V:

Geometric Camera Models: Elements of Analytical Euclidean Geometry, Camera Parameters and the Perspective Projection, Affine Cameras and Affine Projection Equations

Geometric Camera Calibration: Least-Squares Parameter Estimation, A Linear Approach to Camera Calibration, Taking Radial Distortion into Account, Analytical Photogrammetry,

Text Books:

1. David A. Forsyth and Jean Ponce: Computer Vision – A Modern Approach, PHI Learning (Indian Edition), 2009.

Reference Books:

1. E. R. Davies: Computer and Machine Vision – Theory, Algorithms and Practicalities, Elsevier (Academic Press), 4th edition, 2013.
2. R. C. Gonzalez and R. E. Woods "Digital Image Processing" Addison Wesley 2008. 3. Richard Szeliski "Computer Vision: Algorithms and Applications" Springer-Verlag London Limited 2011.

IV Year I Semester	Blockchain Technology (Professional Elective-IV)	L	T	P	C
		3	0	0	3

Course Objectives:

1. To learn the fundamentals of Block Chain and various types of block chain and consensus mechanism.
2. To understand public block chain system, Private block chain system and consortium block chain.
3. Able to know the security issues of blockchain technology.

UNIT- – I:

Fundamentals of Blockchain: Introduction, Origin of Blockchain, Blockchain Solution, Components of Blockchain, Block in a Blockchain, The Technology and the Future.

Blockchain Types and Consensus Mechanism: Introduction, Decentralization and Distribution, Types of Blockchain, Consensus Protocol.

Cryptocurrency: Bitcoin, Altcoin and Token: Introduction, Bitcoin and the Cryptocurrency, Cryptocurrency Basics, Types of Cryptocurrencies, Cryptocurrency Usage.

UNIT- – II:

Public Blockchain System: Introduction, Public Blockchain, Popular Public Blockchains, The Bitcoin Blockchain, Ethereum Blockchain.

Smart Contracts: Introduction, Smart Contract, Characteristics of a Smart Contract, Types of Smart Contracts, Types of Oracles, Smart Contracts in Ethereum, Smart Contracts in Industry.

UNIT- – III:

Private Blockchain System: Introduction, Key Characteristics of Private Blockchain, Private Blockchain, Private Blockchain Examples, Private Blockchain and Open Source, E- commerce Site Example, Various Commands (Instructions) in E-commerce Blockchain, Smart Contract in Private Environment, State Machine, Different Algorithms of Permissioned Blockchain, Byzantine Fault, Multichain.

Consortium Blockchain: Introduction, Key Characteristics of Consortium Blockchain, Need of Consortium Blockchain, Hyperledger Platform, Overview of Ripple, Overview of Corda.

Initial Coin Offering: Introduction, Blockchain Fundraising Methods, Launching an ICO, Investing in an ICO, Pros and Cons of Initial Coin Offering, Successful Initial Coin Offerings, Evolution of ICO, ICO Platforms.

UNIT- – IV:

Security in Blockchain: Introduction, Security Aspects in Bitcoin, Security and Privacy Challenges of Blockchain in General, Performance and Scalability, Identity Management and Authentication, Regulatory Compliance and Assurance, Safeguarding Blockchain Smart Contract (DApp), Security Aspects in Hyperledger Fabric.

Applications of Blockchain: Introduction, Blockchain in Banking and Finance, Blockchain in Education, Blockchain in Energy, Blockchain in Healthcare, Blockchain in Real-estate, Blockchain in Supply Chain, The Blockchain and IoT. Limitations and Challenges of Blockchain.

UNIT- – V:

Blockchain Case Studies:

Case Study 1 – Retail,

Case Study 2 – Banking and Financial Services,

Case Study 3 – Healthcare,

Case Study 4 – Energy and Utilities.

Text book:

1. "Block chain Technology", Chandramouli Subramanian, Asha A.George, Abhilasj K A and Meena Karthikeyan , Universities Press.

Reference Books:

1. Blockchain Blue print for Economy, Melanie Swan, SPD Oreilly.
2. Blockchain for Business, Jai Singh Arun, Jerry Cuomo, Nitin Gauar, Pearson Addition Wesley

IV Year I Semester	AGILE METHODOLOGIES (Professional Elective-V)	L	T	P	C
		3	0	0	3

Course Objectives:

The course aims to:

1. Introduce the fundamentals of **Agile values, principles**, and the Agile Manifesto.
2. Explain how Agile principles are applied to **project planning, delivery**, and **team collaboration**.
3. Provide practical understanding of **Scrum framework** and its core roles, artifacts, and events.
4. Explore **Extreme Programming (XP)** practices, feedback loops, and engineering excellence.
5. Illustrate the use of **Lean thinking, Kanban**, and the role of **Agile coaching** in facilitating organizational change.

Course Outcomes:

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

1. **Explain** Agile fundamentals, values, principles, and the rationale behind the Agile Manifesto.
2. **Apply** Agile practices in real-world projects for effective collaboration, execution, and continuous improvement.
3. **Demonstrate** the use of Scrum roles, events, and artifacts for iterative project management.
4. **Implement** XP practices such as refactoring, continuous integration, and incremental design to enhance code quality.
5. **Analyze** Lean and Kanban approaches to optimize workflow and facilitate Agile transformation through coaching.

UNIT I:

Introduction to Agile-Learning Agile: Getting Agile into Your Brain, Understanding Agile Values and Principles, No Silver Bullet – Agile to the Rescue, Adding Agile Makes a Difference, Fractured Perspectives and Project Failures, The Agile Manifesto and Purpose Behind Practices, Individuals and Interactions over Processes and Tools, Working Software over Comprehensive Documentation, Customer Collaboration over Contract Negotiation, Responding to Change over Following a Plan, Principles over Practices, Understanding Methodologies and Where to Start

UNIT II:

Agile Principles and Project Delivery-The 12 Agile Principles, Customer Collaboration and Clarifying Expectations, Agile in Action: eBook Reader Project Case Study, Communication and Team Coordination, Effective Project Execution and Work Environment, Continuous Improvement for Team and Product, Integrating Agile Principles in the Project Lifecycle

UNIT III:

Scrum and Agile Project Planning-Scrum Fundamentals and Roles, Scrum Master, Product Owner, Development Team, Scrum Values and Events, Daily Scrum, Feedback Loops, Retrospectives, Sprint Planning and Execution, Self-Organizing Teams and Collective Ownership, User Stories, Velocity, Story Points, Burndown Charts, Sprint Boards and Task Management, Cultural Compatibility and Scrum Success

UNIT IV:

Extreme Programming (XP) and Design Practices-XP Practices: Programming, Planning, Integration, Team, Embracing Change through XP Values and Mindset, XP Principles and Feedback Loops, Simplicity and Incremental Design, Identifying Code Smells and Refactoring, Continuous Integration and Avoiding Technical Debt, Collaborative Teams and Holistic Practices

UNIT V:

Lean, Kanban, and Agile Coaching-Lean Thinking and Eliminating Waste, Value Stream Mapping and Root Cause Analysis, Set-Based Development and Flow Efficiency, Delivering Fast and Managing Bottlenecks, Kanban Principles and Visualizing Workflow, Limiting Work in Progress and Measuring Flow, Agile Coaching Principles and Change Facilitation

Textbook:

1. Andrew Stellman & Jill Alison Hart, *Learning Agile*, O'Reilly, 2015

Reference Books:

1. Andrew Stellman & Jennifer Greene, *Head First Agile*, O'Reilly, 2017,
2. Rubin K., *Essential Scrum: A Practical Guide to the Most Popular Agile Process*, Addison-Wesley, 2013

IV Year I Semester	BIG DATA ANALYTICS (Professional Elective-V)	L	T	P	C
		3	0	0	3

Course Objectives:

The course is designed to:

1. Introduce the concept of **Big Data**, its real-world applications across industries, and the enabling technologies like Hadoop and cloud platforms.
2. Explain the fundamentals of **NoSQL databases**, their data models, replication strategies, and use of distributed storage systems like Cassandra.
3. Provide hands-on knowledge in **Hadoop architecture**, HDFS, MapReduce programming, and query processing with Hive.
4. Explore the **Apache Spark framework** for in-memory data processing, transformations, and data analytics.
5. Understand **real-time data processing** using Spark Streaming, structured streaming concepts, and performance tuning strategies.

Course Outcomes:

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

1. **Identify and analyze** Big Data applications and technologies used in various domains like healthcare, finance, and marketing.
2. **Design and implement** NoSQL data models using key-value stores, document stores, and graph databases with tools like Cassandra.
3. **Develop** Big Data processing solutions using Hadoop components such as HDFS, MapReduce, and Hive for structured and semi-structured data.
4. **Utilize** Apache Spark for high-performance, distributed in-memory computing with RDDs, DataFrames, and advanced transformations.
5. **Apply** stream processing concepts in Spark for handling real-time data, including event-time processing, windowing, and watermarks.

UNIT - I:

Big data, convergence of key trends, unstructured data, industry examples, web analytics, marketing, fraud detection, risk analysis, credit risk management, algorithmic trading, healthcare, medicine, advertising, Hadoop, open source technologies, cloud computing, mobile business intelligence, crowdsourcing analytics, inter and trans firewall analytics

UNIT- II:

NoSQL, aggregate data models, key-value data model, document data model, relationships, graph databases, schemaless databases, materialized views, distribution models, sharding, master-slave replication, peer-to-peer replication, consistency, version stamps, Cassandra, table creation, data loading, data reading

UNIT- III:

data formats, Hadoop, data analysis, scaling out, HDFS architecture, fault tolerance, data replication, high availability, data locality, MapReduce architecture, process flow,

Java interface, data flow, Hadoop I/O, data integrity, compression, serialization, Hive, data types, file formats, HiveQL DDL, HiveQL DML, logical joins, window functions, optimization, table partitioning, bucketing, indexing, join strategies

UNIT- IV:
Apache Spark, advantages over Hadoop, lazy evaluation, in-memory processing, DAG, SparkContext, SparkSession, RDD, narrow transformations, wide transformations, actions, DataFrames, RDD to DataFrame conversion, Catalyst optimizer, transformations on DataFrames, working with dates, working with timestamps, handling nulls, complex types, JSON, grouping, window functions, joins, data sources, broadcast variables, accumulators, on-premises deployments, cluster managers, standalone mode, Spark on YARN, Spark logs, Spark UI, Spark UI history server, debugging, Spark first aid

UNIT- V:
Spark performance tuning, stream processing, event-time processing, stateful processing, tumbling windows, late data handling, watermarks, duplicate removal, structured streaming, streaming core concepts, structured streaming in action, stream transformations, stream input, stream output

Text Books:

1. Big Data, Big Analytics: Emerging, Michael Minnelli, Michelle Chambers, and AmbigaDhiraj, 1st edition ,2013
2. SPARK: The Definitive Guide, Bill Chambers &MateiZaharia, O'Reilley, 2018-first Edition.
3. Business Intelligence and Analytic Trends for Today's Businesses", Wiley, First edition-2013.
4. P. J. Sadalage and M. Fowler, "NoSQL Distilled: A Brief Guide to the Emerging World Polyglot Persistence", Addison-Wesley Professional, 2012
5. Tom White, "Hadoop: The Definitive Guide", Third Edition, O'Reilley, 2012

Reference Books:

1. "Hadoop Operations", O'Reilley, Eric Sammer, First Edition -2012.
2. "Programming Hive", O'Reilley, E. Capriolo, D. Wampler, and J. Rutherglen, 2012.
3. "HBase: The Definitive Guide", O'Reilley, Lars George, September 2011: First Edition..
4. "Cassandra: The Definitive Guide", O'Reilley, Eben Hewitt, 2010.
5. "Programming Pig", O'Reilley, Alan Gates, October 2011: First Edition

IV Year I Semester	MOBILE COMPUTING (Professional Elective-V)	L	T	P	C
		3	0	0	3

Course Objectives:

The course is designed to:

1. Introduce the fundamentals of **mobile communication technologies**, their standards, and architecture.
2. Explain **GSM, GPRS, CDMA, 3G, and 4G architectures** and protocols used in mobile systems.
3. Provide insight into **mobile IP**, **ad hoc networks**, and **wireless sensor networks**.
4. Explore **data synchronization** and **mobile agent-based computing** for mobile environments.
5. Study **short-range wireless networks**, **WLAN**, **WAP**, and **mobile internet protocols**.

Course Outcomes:

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

1. **Describe** mobile communication techniques, architectures, and limitations of mobile systems.
2. **Compare and contrast** various mobile communication standards including GSM, CDMA, 3G, and 4G.
3. **Analyze** mobile network layer functionalities, mobile IP, MANETs, and wireless sensor networks.
4. **Apply** synchronization methods and mobile agent principles in mobile application development.
5. **Evaluate** WLAN architecture, WAP protocol stack, and mobile internet communication mechanisms.

UNIT--I:

Mobile Communications: An Overview- Mobile Communication-guided transmission, unguided transmission- signal propagation frequencies, antennae, modulation, modulation methods and standards for voice-oriented data communication standards, modulation methods and standards for data and voice communication, mobile computing- novel applications and limitations, mobile computing architecture, mobile system networks. Mobile devices and systems: Cellular networks and frequency reuse, Mobile smart phones, Smart mobiles and systems, handheld pocket computers, Handheld devices, Smart systems, Limitations of mobile devices

UNIT--II:

GSM and other 2G Architectures: GSM-services and system architecture, Radio interfaces of GSM, Protocols of GSM, Localization, Call handling, GPRS system architecture. Wireless medium access control, CDMA, 3G,and 4G

Communication: Modulation, Multiplexing, Controlling the medium access, Spread spectrum, Coding methods, IMT-20003G wireless communication standards, WCDMA 3G communication standards, CDMA 3G communication standards, Broadband wireless access, 4Gnetworks.

UNIT--III:

Mobile IP Network layer: IP and Mobile IP network layers: OSI layer functions, TCP/IP and Internet protocol, Mobile internet protocol; Packet delivery and Handover Management;

Location Management: Agent Discovery; Mobile TCP

Introduction to Mobile Adhoc network: fixed infrastructure architecture, MANET infrastructure architecture;

MANET: properties, spectrum, applications; Security in Ad-hoc network; Wireless sensor networks; sensor network applications.

UNIT--IV:

Synchronization: Synchronization in mobile computing systems, Usage models for Synchronization in mobile application, Domain-dependant specific rules for data synchronization, Personal information manager, synchronization and conflict resolution strategies, synchronizer; Mobile agent: mobile agent design, aglets; Application Server

UNIT--V:

Mobile Wireless Short Range Networks and Mobile Internet: Wireless networking and wireless LAN, Wireless LAN (WLAN) architecture, IEEE 802.11protocol layers, Wireless application protocol (WAP)-WAP1.1 architecture, wireless datagram protocol (WDP), Wireless Transport Layer Security (WTLS), wireless transaction and session layers, wireless application environment.

TEXTBOOK:

1. RAJ KAMAL, "Mobile Computing," second edition, Oxford.

2. ASOKE K TALUKDER, HASANAHMED, ROOPA R YAVAGAL, "Mobile Computing, Technology Applications and Service Creation" Second Edition, Mc Graw Hill.

3. UWE Hansmann, Lother Merk, Martin S. Nocklous, Thomas Stober, "Principles of Mobile Computing," Second Edition, Springer

IV Year I Semester	CYBER PHYSICAL SYSTEMS (Professional Elective-V)	L	T	P	C
		3	0	0	3

Course Objectives:

This course is designed to:

1. Introduce **symbolic synthesis techniques** and **symbolic model construction** for designing cyber-physical systems (CPS).
2. Explain the **security requirements**, attack models, and countermeasures for securing CPS.
3. Familiarize students with **synchronization and consensus algorithms** in distributed CPS architectures.
4. Provide insights into **real-time scheduling techniques** including multiprocessor and memory-aware scheduling for CPS.
5. Teach students about **model integration** and formal methods for semantics and language integration in CPS modeling.

Course Outcomes:

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

1. **Construct symbolic models** for cyber-physical systems using synthesis techniques and relevant software tools.
2. **Analyze and design secure CPS architectures** by identifying security threats and proposing system-theoretic countermeasures.
3. **Apply synchronization and consensus algorithms** in distributed CPS and explain the role of formal software engineering.
4. **Implement real-time scheduling algorithms** that accommodate timing variability and resource constraints in multicore environments.
5. **Integrate models and semantics** using domain-specific modeling languages (DSMLs) and formal language integration methods for CPS design.

UNIT- I:

Symbolic Synthesis for Cyber-Physical Systems: Introduction and Motivation, Basic Techniques - Preliminaries, Problem Definition, Solving the Synthesis Problem, Construction of Symbolic Models, Advanced Techniques: Construction of Symbolic Models, Continuous-Time Controllers, Software Tools

UNIT- II:

Security of Cyber-Physical Systems: Introduction and Motivation, Basic Techniques - Cyber Security Requirements, Attack Model, Countermeasures, Advanced Techniques: System Theoretic Approaches

UNIT- III:

Synchronization in Distributed Cyber-Physical Systems: Challenges in Cyber-Physical Systems, A Complexity-Reducing Technique for Synchronization, Formal Software Engineering, Distributed Consensus Algorithms, Synchronous Lockstep Executions, Time-Triggered Architecture, Related Technology, Advanced Techniques

UNIT- IV:

Real-Time Scheduling for Cyber-Physical Systems: Introduction and Motivation, Basic Techniques - Scheduling with Fixed Timing Parameters, Memory Effects, Multiprocessor/Multicore Scheduling, Accommodating Variability and Uncertainty

UNIT- V:

Model Integration in Cyber-Physical Systems: Introduction and Motivation, Causality, Semantic Domains for Time, Interaction Models for Computational Processes, Semantics of CPS DSMLs, Advanced Techniques, ForSpec, The Syntax of CyPhyML, Formalization of Semantics, Formalization of Language Integration.

Text Books:

1. Raj Rajkumar, Dionisio De Niz, and Mark Klein, Cyber-Physical Systems, Addison-Wesley Professional, 2016
2. Rajeev Alur, Principles of Cyber-Physical Systems, MIT Press

Reference Books:

1. E.A.Lee, Sanjit Seshia, Introduction to Embedded Systems: A Cyber-Physical Systems Approach, MIT Press
2. Andre Platzer, Logical Foundations of Cyber-Physical Systems, (2e), Springer Publishing, 2018

IV Year I Semester	CLOUD COMPUTING (Professional Elective-V)	L	T	P	C
		3	0	0	3

Course Objectives

This course aims to:

1. Introduce the **fundamentals and service models** of cloud computing and its **deployment architectures**.
2. Explain the **enabling technologies** such as distributed computing, virtualization, and service-oriented architectures.
3. Familiarize students with **virtual machines, containers, and orchestration platforms** like Docker and Kubernetes.
4. Discuss **key challenges** in cloud computing including **security, scalability, interoperability, and energy efficiency**.
5. Expose learners to **advanced cloud topics** including serverless computing, IoT-cloud integration, edge/fog computing, and DevOps.

Course Outcomes

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

1. **Describe the architecture and service models** of cloud computing and explain the benefits and challenges of different deployment models.
2. **Apply distributed and parallel computing concepts** and demonstrate understanding of enabling technologies such as RPC, SOA, and Web services.
3. **Analyze and differentiate between virtualization and containerization technologies**, and implement orchestration using Kubernetes or Docker Swarm.
4. **Evaluate cloud-specific issues** including interoperability, scalability, fault tolerance, and security architectures.
5. **Explore and apply advanced concepts** like serverless computing, IoT-cloud interaction, edge/fog computing, and DevOps practices in cloud environments.

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, 2nd edition, Rajkumar Buyya, Christian Vecchiola, Thamarai Selvi, Shivananda Poojara, Satish N. Srirama, Mc Graw Hill, 2024.
2. Distributed and Cloud Computing, Kai Hwang, Geoffery C. Fox, Jack J. Dongarra, Elsevier, 2012.

Reference Books:

1. Cloud Computing, Theory and Practice, Dan C Marinescu, 2nd 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)

IV Year I Semester	PROMPT ENGINEERING (Skill Enhancement Course)	L	T	P	C
		0	1	2	2

Course Objectives:

The main objectives of the course are to

i. Apply iterative prompting for clarity and context.
ii. Create varied prompts to steer model outputs.
iii. Construct chain-of-thought and structured prompts.
iv. Develop retrieval-augmented pipelines to ground outputs.
v. Evaluate LLM agents and multimodal apps for ethics and robustness.

Unit I: Foundations of Prompt Engineering: Definition of prompt engineering, Distinction between prompt engineering and model fine-tuning, Motivation and benefits of prompt engineering, Core principles of effective prompt design**,** Anatomy of a prompt, Setting up the Python environment for LLM interaction, Iterative prompting lifecycle, Common prompt pitfalls and remediation

Lab Experiments:

1. Environment & Connectivity: Install required packages (e.g., transformers, openai); securely configure the API key; run a simple "Hello, world" prompt to verify model access.
2. Baseline vs. Enhanced Prompts: Execute a naïve prompt ("Write a one-paragraph bio of Ada Lovelace.") and an enhanced prompt that adds role framing, specificity, and explicit format instructions; compare both outputs for relevance, completeness, and style.
3. Iterative Refinement on a Simple Task: Summarize the plot of the Shakespearean play Romeo and Juliet in two sentences through three rounds of prompt tweaking:
    a. Minimal instruction.
    b. Addition of length and style constraints
    c. Specification of key content elements (setting and theme)

    Document how each iteration changes and improves the result.
4. Diagnosing Prompt Failures & Edge Cases: Craft a vague or contradictory prompt; analyze the failure mode (ambiguity, missing context, or format errors); refine the prompt by adding examples or clarifying instructions.

Unit II: Advanced Prompt Patterns & Techniques: Enhanced prompt anatomy: contextual detail and explicit output specifications, Few-shot in-context prompting, Prompt structuring and template design, Role-based prompting to establish personas or system behavior, Negative prompting to filter or suppress undesired content, Constraint specification and instruction enforcement (e.g., length, format), Iterative prompt refinement and optimization

Lab Experiments:

1. Few-Shot vs. Zero-Shot Comparison: Design and execute a zero-shot prompt and a few-shot prompt (with 2–3 exemplar input-output pairs) for a chosen text task (e.g., sentiment classification or translation); compare outputs for accuracy, consistency, and adherence to examples.
2. Role-Based & Negative Prompting: Craft a role-based prompt to establish a specific persona (e.g., "You are a financial advisor…"); then create a negative prompt to suppress undesired content (e.g., "Do not mention any brand names"); evaluate how each influences the model's response.
3. Constraint Specification & Iterative Refinement: Select an open-ended task (e.g., summarizing a technical article); issue a basic prompt; identify failures in length or format; refine the prompt by adding explicit constraints (word count, bullet format, etc.); document improvements over two refinement cycles.

Unit III: Structured Output & Reasoning Techniques: Importance of structured outputs for real-world applications, Prompting for specific formats (lists, tables, Markdown), Generating valid JSON and YAML via explicit instructions, Eliciting chain-of-thought reasoning in zero-shot prompts, Decomposing complex tasks into manageable sub-tasks

Lab Experiments:

1. Structured Format Prompting: Instruct the model to output information as bullet lists and Markdown tables (e.g., "List three benefits of daily exercise in a Markdown table with columns 'Benefit' and 'Description.'"); verify the output matches the requested structure.
2. JSON/YAML Generation: Provide a brief dataset description (e.g., three books with title, author, publication year) and prompt the model to produce valid JSON or YAML; use a parser to validate syntax and refine the prompt if errors occur.
3. Chain-of-Thought & Task Decomposition: Present a multi-step problem (e.g., a logic puzzle) and apply zero-shot CoT prompting (e.g., "Let's think step by step. Explain your reasoning before the final answer."); separately, decompose the problem into sequential sub-questions, collect partial answers, combine them, and compare accuracy against a direct-answer baseline.

Unit IV: Retrieval-Augmented Generation & LangChain Workflows: Limitations of LLM internal knowledge, Need for external data sources, Introduction to Retrieval-Augmented Generation (RAG), Overview of RAG architecture (indexing vs. retrieval + generation), Getting started with LangChain for LLM applications, Basics of LangChain Expression Language (LCEL), Simplified indexing pipeline: document loading & text splitting, Fundamentals of embeddings and vector stores, Building a basic retrieval-generation pipeline with an LCEL chain

Lab Experiments:

1. Building a Simple LCEL Chain: Create a minimal LCEL script that accepts a fixed instruction (e.g., "Summarize this text: …"), passes it to an LLM, and prints the result; verify end-to-end execution.
2. Basic Data Indexing for RAG: Load a small collection of documents; split into uniform chunks (e.g., 200 tokens); generate embeddings for each chunk; store them in an in-memory vector store; inspect for consistency.
3. Constructing & Running a Basic RAG Chain: Build a pipeline that:
    a. Receives a user query
    b. Retrieves the top-k relevant chunks
    c. Constructs a combined prompt with context + query
    d. Send it to the LLM
    e. Returns the answer

Test with sample queries and compare factual accuracy against a prompt without retrieval.

Unit V: Agents, Multimodal AI & Ethical Evaluation: Introduction to LLM agents and their basic architecture, Overview of multimodal AI models (VLMs), Prompting for text-to-image generation and image understanding, Importance of prompt evaluation beyond subjective judgment, Manual evaluation techniques (heuristic checks for accuracy, relevance, format), Introduction to "LLM-as-Judge" for automated evaluation, Security considerations (prompt injection, sensitive-information risks), Prompt-based mitigation strategies for safety and robustness, Ethical concerns (bias, misinformation, data privacy), Brief exploration of UI frameworks (Streamlit/Gradio) for deploying prompt-driven apps, Adapting to the evolving nature of prompt engineering through continuous learning

Lab Experiments:

1. Building a Simple LLM Agent: Register a tool (e.g., a calculator function) and craft prompts that instruct the agent to invoke it when required; implement using Lang Chain or a function-calling API; test on queries requiring tool execution.
2. Multimodal Prompting Exploration: Generate images from detailed text prompts; feed one generated image into an image-understanding model or API with an appropriate prompt; compare the returned caption to the original prompt to evaluate alignment.
3. Prompt Evaluation & Ethics Workshop:
    a. Select two existing prompts and generate multiple outputs; apply manual heuristic checks for accuracy, relevance, and format compliance.
    b. Use an "LLM-as-Judge" prompt (e.g., "Rate these outputs on a scale of 1–5 for clarity and correctness.") to automate evaluation.
    c. Design a prompt- injection test (e.g., "Ignore previous instructions…"), observe the response, then refine system prompts to mitigate the vulnerability.

IV Year I Semester	CONSTITUTION OF INDIA	L	T	P	C
		2	0	0	-

Course Objectives:

- Understand the premises informing the twin themes of liberty and freedom from a civil rights perspective.
- To address the growth of Indian opinion regarding modern Indian intellectuals' constitutional role and entitlement to civil and economic rights as well as the emergence of nationhood in the early years of Indian nationalism.
- To address the role of socialism in India after the commencement of the Bolshevik Revolution in 1917 and its impact on the initial drafting of the Indian Constitution.

UNIT–I:

History of Making of the Indian Constitution: History, Drafting Committee, (Composition & Working)

Philosophy of the Indian Constitution- Preamble, Salient, Features

UNIT–II:

Contours of Constitutional Rights & Duties: Fundamental Rights, Right to Equality, Right to Freedom, Right against Exploitation, Right to Freedom of Religion, Cultural and Educational Rights, Right to Constitutional Remedies, Directive Principles of State Policy, Fundamental Duties.

UNIT–III:

Organs of Governance: Parliament, Composition, Qualifications and Disqualifications, Powers and Functions, **Executive-** President, Governor, Council of Ministers, Judiciary, Appointment and Transfer of Judges, Qualifications, Powers and Functions

UNIT–IV:

Local Administration: District's Administration head: Role and Importance, Municipalities: Introduction, Mayor and role of Elected Representative CEO of Municipal Corporation, Pachayati raj: Introduction, PRI: ZilaPachayat, Elected officials and their roles, CEO ZilaPachayat: Position and role, Block level: Organizational Hierarchy (Different departments), Village level: Role of Elected and Appointed officials, Importance of grass root democracy

UNIT–V:

Election Commission: Election Commission: Role and Functioning, Chief Election Commissioner and Election Commissioners, State Election Commission: Role and Functioning, Institute and Bodies for the welfare of SC/ST/OBC and women.

Text Books:

1. The Constitution of India, 1st Edition, (Bare Act), Government Publication, 1950
2. Framing of Indian Constitution, 1st Edition, Dr. S. N. Busi, Dr. B. R. Ambedkar 2015

Reference Books:

1. Indian Constitution Law, 7th Edition, M. P. Jain, Lexis Nexis, 2014

JNTUGV- COLLEGE OF ENGINEERING VIZIANAGARAM

DEPARTMENT OF INFORMATION TECHNOLOGY

B.TECH- IT (R23)-HONORS

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY GURJADA VIZIANAGARAM
VIZIANAGARAM – 535 003, Andhra Pradesh, India
COLLEGE OF ENGINEERING VIZIANAGARAM(A)
B.Tech IT (R23-COURSE STRUCTURE)

HONOR DEGREE IN INFORMATION TECHNOLOGY

(I) Cyber Security

S.No	Subject Title	L	T	P	C
1	Cyber Security Essentials	3	0	0	3
2	Secure Coding	3	0	0	3
3	Vulnerability Assessment & Penetration Testing	3	0	0	3
4	Malware Analysis	3	0	0	3
5	03 MOOCS courses @ 2credits each (8 weeks course) 02 MOOCS courses @ 3credits each (12 weeks course) (Any CSE/IT related **Program Core subject** from NPTEL/ SWAYAM course other than the courses listed above needs to be taken)				6
				Total	**18**

(II) AI/ML

S.No	Subject Title	L	T	P	C
1	Statistical Machine Learning	3	0	0	3
2	Reinforcement Learning	3	0	0	3
3	Programming Foundations with GenAI Tools	3	0	0	3
4	Advanced Computer Vision	3	0	0	3
5	03 MOOCS courses @ 2credits each (8 weeks course) 02 MOOCS courses @ 3credits each (12 weeks course) (Any CSE/IT related **Program Core subject** from NPTEL/ SWAYAM course other than the courses listed above needs to be taken)				6
				Total	**18**

(III) Data Science

S.No	Subject Title	L	T	P	C
1	Mathematical Essential for Data Science	3	0	0	3
2	Introduction to Data Science	2	0	2	3
3	Data Analytics and Visualization	2	0	2	3
4	Graph Analytics	2	0	2	3
5	03 MOOCS courses @ 2credits each (8 weeks course) 02 MOOCS courses @ 3credits each (12 weeks course) (Any CSE/IT related **Program Core subject** from NPTEL/ SWAYAM course other than the courses listed above needs to be taken)				6
				Total	**18**

Note: Students who have registered for honors program can opt a maximum of two (02) courses per semester.

	Honor Course	L	T	P	C
		3	1	0	4
CYBER SECURITY ESSENTIALS					

Course Objective:

1. To introduce information security concepts to undergraduate engineering students, so they can defend their personal and organizational information from probable security attacks and incidents.
2. Classify and analyze various categories of network attacks
3. Describe and compare various cryptographic algorithms

Course Outcomes:

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

1. Understand the basics and need for information security
2. Identify, analyze, and evaluate infrastructure and network vulnerabilities.
3. Understand and analyze different access control and authentication methods.
4. Identify and assess current and anticipated security risks and vulnerabilities with vulnerability assessment and auditing methods.
5. Learn the fundamentals of cryptography and how cryptography serves as the central language of information security.

UNIT-I:

Introduction to Security: Challenges of Securing Information, Definition of Information Security, Attackers, Attacks and Defenses.

Systems Threats and Risks: Software-Based Attacks, Hardware-Based Attacks, Attacks on Virtualized Systems, Hardening the Operating System, Preventing Attacks that Target the Web Browser, Hardening Web Servers, Protecting Systems from Communications-Based Attacks, Applying Software Security Applications.

UNIT-II:

Network Vulnerabilities and Attacks: Network Vulnerabilities, Categories of Attacks, Methods of Network Attacks.

Network Defenses: Crafting a Secure Network, Applying Network Security Devices, Host and Network Intrusion Prevention Systems (HIPS/NIPS), Protocol Analyzers, Internet Content Filters, Integrated Network Security Hardware.

UNIT-III:

Access Control: Access Control Models and Practices, Logical Access Control Methods, Physical Access Control.

Authentication: Definition of Authentication, Authentication Credentials, Extended Authentication Protocols, Remote Authentication and Security.

UNIT-IV:

Vulnerability Assessment: Risk Management, Assessment, and Mitigation, Identifying Vulnerabilities.

Security Audit: Privilege Auditing, Usage Auditing, Monitoring Methodologies and Tools.

UNIT-V:

Cryptography: Introduction to Cryptography, Cryptographic Algorithms, Using Cryptography on Files and Disks, Digital Certificates, Public Key Infrastructure, Key Management.

Text Book:

1. Security+ Guide to Network Security Fundamentals, Third Edition, Mark Ciampa, Cengage Learning.

Reference Books :

1. Principles of Information Security, Michael E. Whitman and Herbert J. Mattord, Cengage Learning.
2. Information Security: The Complete Reference, Rhodes-Ousley, Mark, Second Edition, McGraw-Hill.
3. Information Security: Principles and Practices, Mark S. Merkow, Jim Breithaupt, 2nd Edition, Pearson Education.

	Honor Course	L	T	P	C
		3	0	2	4
SECURE CODING					

Course Objectives:

1. Understanding of the various security attacks and knowledge to recognize and remove common coding errors that lead to vulnerabilities.
2. Knowledge of outline of the techniques for developing a secure application.
3. Recognize opportunities to apply secure coding principles.

Course Outcomes:

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

1. List of secure systems and various security attacks
2. Demonstrate the development of process of software leads to secure coding practices
3. Apply Secure programs and various risk in the software's
4. Classify various errors that lead to vulnerabilities
5. Design Real time software and vulnerabilities

UNIT–I:

Introduction-Need for secure systems, Proactive security development process, Security principles to live by and threat modeling.

UNIT–II:

Secure Coding in C-Character strings- String manipulation errors, String Vulnerabilities and exploits Mitigation strategies for strings, Pointers, Mitigation strategies in pointer based vulnerabilities Buffer Overflow based vulnerabilities

UNIT–III:

Secure Coding in C++ and Java-Dynamic memory management, Common errors in dynamic memory management, Memory managers, Double –free vulnerabilities, Integer security, Mitigation strategies

UNIT–IV:

Database and Web Specific Input Issues-Quoting the Input, use of stored procedures, Building SQL statements securely, XSS related attacks and remedies

UNIT–V:

Software Security Engineering-Requirements engineering for secure software: Misuse and abuse cases, SQUARE process model Software security practices and knowledge for architecture and design

Text Book:

1. Michael Howard, David LeBlanc, "Writing Secure Code", Microsoft Press, 2nd Edition, 2003.

References:

1. Robert C. Seacord, "Secure Coding in C and C++", Pearson Education, 2nd edition, 2013.
2. Julia H. Allen, Sean J. Barnum, Robert J. Ellison, Gary McGraw, Nancy R. Mead, "Software Security Engineering: A guide for Project Managers", Addison-Wesley Professional, 2008.

	Honor Course	L	T	P	C
		3	1	0	4
VULNERABILITY ASSESSMENT & PENETRATION TESTING					

Course Objectives:

1. To identify security vulnerabilities and weaknesses in the target applications.
2. To identify how security controls can be improved to prevent hackers gaining access to operating systems and networked environments.
3. To test and exploit systems using various tools.

Course Outcomes:

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

1. Explain Penetration testing phases
2. Illustrate information gathering methodologies
3. Apply System Hacking Techniques in real time applications
4. Explore advanced System hacking
5. Describe Bypassing WLAN Authentication

UNIT-I:

Introduction-Penetration Testing phases/Testing Process, types and Techniques, Blue/Red Teaming, Strategies of Testing, Non-Disclosure Agreement Checklist, Phases of hacking, Open-source/proprietary Pentest Methodologies

UNIT-II :

Information Gathering and Scanning- Information gathering methodologies- Foot printing, Competitive Intelligence- DNS Enumerations- Social Engineering attacks, Port Scanning-Network Scanning- Vulnerability Scanning- NMAP scanning tool- OS Fingerprinting-Enumeration.

UNIT-III :

System Hacking - Password cracking techniques- Key loggers- Escalating privileges- Hiding Files, Double Encoding, Steganography technologies and its Countermeasures. Active and passive sniffing- ARP Poisoning, MAC Flooding- SQL Injection - Error- based, Union-based, Time-based, Blind SQL, Out-of-band. Injection Prevention Techniques.

UNIT- IV :

Advanced System Hacking- Broken Authentication, Sensitive Data Exposure, XML External Entities, Broken Access Code, XSS - Stored, Reflected, DOM Based

UNIT-V :

Wireless Pen test: Wi-Fi Authentication Modes, Bypassing WLAN Authentication, Types of Wireless Encryption, WLAN Encryption Flaws, AP Attack, Attacks on the WLAN Infrastructure, DoS-Layer1, Layer2, Layer 3, DDoS Attack, Client Misassociation, Wireless

Hacking Methodology, Wireless Traffic Analysis

Text Books:

1. Kali Linux 2: Windows Penetration Testing, By Wolf Halton, Bo Weaver , June 2016 Packt Publishing

References:

1. Mastering Modern Web Penetration Testing By Prakhar Prasad,October 2016 Packt Publishing.
2. SQL Injection Attacks and Defense 1st Edition, by Justin Clarke-Salt, Syngress Publication

	Honor Course	L	T	P	C
		3	0	2	4
MALWARE ANALYSIS					

Course Objectives:

1. To understand the purpose of computer infection program.
2. To implement the covert channel and mechanisms.
3. To test and exploit various malware in open-source environment.

Course Outcomes:

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

1. Explain the characteristics of Malware and its effects on Computing systems.
2. Predict the given system scenario using the appropriate tools to Identify the vulnerabilities and to perform Malware analysis.
3. Analyze the given Portable Executable and Non-Portable Executable files using Static and dynamic analysis techniques.
4. Demonstrate the Malware functionalities.
5. How to apply anti-reverse engineering in different Applications

UNIT–I:

Malware Basics- General Aspect of Computer infection program, Non Self Reproducing Malware, How does Virus Operate, Virus Nomenclature, Worm Nomenclature, Recent Malware Case Studies.

UNIT– II:

Basic Analysis- Antivirus Scanning, x86 Disassembly, Hashing, Finding Strings, Packed Malware, PE File Format, Linked Libraries & Functions, PE Header File &Section.

UNIT–III:

Advanced Static & Dynamic Analysis-IDA Pro, Recognizing C code constructs, Analyzing malicious windows program, Debugging, OllyDbg, Kernel Debugging with WinDbg, Malware Focused Network Signatures.

UNIT–IV:

Malware Functionalities-Malware Behavior, Covert Malware Launch, Data Encoding, Shell code Analysis.

UNIT–V:

Reverse Engineering Malware (REM): REM Methodology, Resources for Reverse-Engineering Malware (REM) Understanding Malware Threats, Malware indicators, Malware Classification, Examining Clam AV-Signatures.

Text Books:

1. Michael Sikorski, Andrew Honig "Practical Malware Analysis: The Hands-On Guide to Dissecting Malicious Software" publisher Williampollock

References:

1. ErciFiliol, "Computer Viruses: from theory to applications", Springer, 1st edition, 2005.

	Honor Course	L	T	P	C
		3	1	0	4
STATISTICAL MACHINE LEARNING					

Course Objectives:

1. To provide foundational knowledge of statistical and probabilistic methods relevant to machine learning.
2. To familiarize students with both probabilistic and non-probabilistic models for learning from data.
3. To develop the ability to evaluate, interpret, and validate machine learning models using statistical metrics.
4. To equip students with the knowledge of advanced and research-relevant machine learning techniques.
5. To introduce key topics in modern machine learning including Bayesian reasoning, ensemble learning, deep generative models, and reinforcement learning.

Course Outcomes:

1. Apply probability and inference techniques in machine learning tasks.
2. Implement and evaluate regression/classification models statistically.
3. Perform Bayesian reasoning and probabilistic model updates.
4. Use kernel and ensemble methods for complex datasets.
5. Explore advanced topics like Bayesian networks, deep generative modeling, and reinforcement learning.

UNIT I:

Introduction and Foundations- What is Machine Learning? ,Types of Learning: Supervised, Unsupervised, Reinforcement, Basics of Statistical Inference, Probability Theory Fundamentals, Random Variables, Expectation & Variance, Bayes' Theorem, Maximum Likelihood Estimation (MLE), Bias-Variance Tradeoff

UNIT II:

Linear Models for Regression and Classification-Linear Regression, Least Squares Estimation, Regularization Techniques (Ridge & Lasso), Logistic Regression, Generalized Linear Models (GLMs), Probabilistic Interpretations, Gradient Descent Optimization, Evaluation Metrics: RMSE, MAE, Accuracy, Precision, Recall, ROC Curve, AUC

UNIT III:

Bayesian Learning and Inference- Bayesian Decision Theory, Prior–Likelihood–Posterior Fundamentals, Maximum A Posteriori (MAP), Naïve Bayes Classification, Gaussian Distribution, Bayesian Linear Regression, Conjugate Priors, Model Evidence, Bayesian Occam's Razor

UNIT IV:

Non-Linear Models and Kernel Methods- Decision Trees, Random Forests, k-Nearest Neighbors, Support Vector Machines, Kernel Trick, Polynomial & RBF Kernels, Bias-Variance in Non-linear Models, Cross-Validation Techniques, Hyperparameter Tuning

UNIT V:

Unsupervised Learning & Dimensionality Reduction- k-Means Clustering, Gaussian Mixture Models (GMM), Expectation-Maximization (EM), Hierarchical Clustering, Principal Component Analysis (PCA), t-SNE, Anomaly Detection Techniques, Model Selection Criteria (AIC, BIC), Intro to Graphical Models

UNIT VI:

Advanced Topics- Markov Chain Monte Carlo (MCMC), Gibbs Sampling, Variational Inference, Ensemble Learning (Bagging, Boosting, Stacking), Gradient Boosting (XGBoost, LightGBM), Gaussian Processes, Bayesian Neural Networks, Deep Generative Models (VAEs, GANs), Information Theory Essentials (Entropy, KL-Divergence, Mutual Info), Semi-Supervised Learning, Active Learning, Multi-Task & Transfer Learning, Meta-Learning, Fairness in ML, Explainable AI (XAI), Causal Inference, Reinforcement Learning (Q-Learning, Policy Gradients)

Textbooks:

1. **Kevin P. Murphy** – *Probabilistic Machine Learning: An Introduction*, **MIT Press**, Volume 1 (Mar 2022)
2. **Christopher M. Bishop** – *Pattern Recognition and Machine Learning*, updated paperback, **BookRivers** (India, Apr 2025)

References:

1. **Kevin P. Murphy** – *Probabilistic Machine Learning: Advanced Topics*, **MIT Press**, Volume 2 (2022)
2. **Tom M. Mitchell** – *Machine Learning*, **McGraw-Hill Education**, foundational text (1997; widely used)
3. **T. Hastie, R. Tibshirani & J. Friedman** – *The Elements of Statistical Learning*, **Springer**, 2nd Ed. (2009)
4. **Shai Shalev-Shwartz & Shai Ben-David** – *Understanding Machine Learning: From Theory to Algorithms*, **Cambridge Univ. Press**, 2014
5. **David Barber** – *Bayesian Reasoning and Machine Learning*, **Cambridge Univ. Press**, 2012

	Honor Course	L	T	P	C
		3	1	0	4
REINFORCEMENT LEARNING					

Course Objectives

1. To introduce the fundamental concepts and mathematical foundations of Reinforcement Learning (RL).
2. To explain the RL framework including agents, environments, states, actions, and rewards.
3. To provide hands-on understanding of classical RL algorithms such as value iteration, Q-learning, and SARSA.
4. To explore the trade-off between exploration and exploitation in decision-making under uncertainty.
5. To introduce function approximation techniques and their integration with deep learning in RL.
6. To study advanced RL techniques including policy gradients, actor-critic methods, and deep reinforcement learning.
7. To expose learners to hierarchical RL, multi-agent settings, and real-world RL applications.
8. To highlight current trends, challenges, evaluation methods, and ethical considerations in the deployment of RL systems.

Course Outcomes:

1. Formulate real-world problems as RL setups (states, actions, rewards).
2. Implement and analyze classical RL algorithms (value iteration, Q-learning, SARSA).
3. Apply exploration–exploitation strategies in bandit and MDP settings.
4. Utilize function approximation and deep neural networks in RL.
5. Explore policy gradient methods and modern deep RL algorithms.

UNIT I:

Foundations of Reinforcement Learning-Introduction to Reinforcement Learning, Elements of RL framework, Multi-armed bandits: k-armed bandit problems, actions-value methods, exploration vs exploitation (UCB, gradient bandit), Markov Decision Processes (MDP), Bellman equations, Dynamic Programming: policy evaluation, policy iteration, value iteration

UNIT II:

Model-Free Learning and Temporal-Difference Methods-Monte Carlo methods (first-visit, every-visit, control), Temporal-Difference Learning (TD(0)), SARSA, Q-Learning, Double Q-Learning, n-step TD methods, eligibility traces (TD(λ)), off-policy vs on-policy learning

UNIT III:

Function Approximation and Deep RL-Linear value-function approximation, semi-gradient TD methods, neural network architectures for RL, Deep Q-Networks (DQN), experience replay, target networks, deep extensions (Double DQN, Dueling DQN), Policy gradient methods: REINFORCE, actor-critic methods, Advantage Actor-Critic (A2C), Deep Deterministic Policy Gradient (DDPG), PPO

UNIT IV:

Hierarchical & Advanced Reinforcement Learning- Extensions: Semi-Markov Decision Processes (semi-MDPs), options framework, hierarchical reinforcement learning, POMDPs, multi-agent RL settings (cooperative and competitive), contextual bandits, trust region methods, recent developments

UNIT V:

Case Studies, Ethics, and Applications-Applications of RL: games (Atari, Go/AlphaGo), robotics, recommendation systems; RL from human feedback (RLHF); challenges like sample complexity, stability, fairness; evaluation metrics: regret, convergence, computational cost; ethical implications and responsible deployment

Textbooks :

1. **Richard S. Sutton & Andrew G. Barto** – *Reinforcement Learning: An Introduction*, 2nd Edition, **MIT Press**, 2018
2. **Dimitri P. Bertsekas** – *Reinforcement Learning and Optimal Control*, 1st Edition, **Athena Scientific**, 2019

References :

1. **Dimitri P. Bertsekas** – *Dynamic Programming and Optimal Control*, Volumes I & II, **Athena Scientific**, 4th Edition, 2017
2. **Csaba Szepesvári** – *Algorithms of Reinforcement Learning*, **Morgan & Claypool**, 2010
3. **Warren B. Powell** – *Reinforcement Learning and Stochastic Optimization*, **Wiley**, 2022
4. **Aske Plaat** – *Deep Reinforcement Learning: A Textbook* (2022 draft), covers advanced topics like hierarchical RL and multi-agent RL

	Honor Course	L	T	P	C
		3	1	0	4
PROGRAMMING FOUNDATIONS WITH GENAI TOOLS					

Course Objectives:

By the end of the course, students will be able to:

1. Understand the basics of programming and the capabilities of modern Generative AI tools in aiding software development.
2. Learn data representation, manipulation, and control structures in Python.
3. Develop skills in code comprehension, testing, and debugging with both human and AI assistance.
4. Compare programming paradigms between Python and C, and understand low-level implementation concepts.
5. Explore performance improvements and gain a preview of advanced programming concepts such as data structures and algorithms.

Course Outcomes:

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

1. Explain the fundamentals of programming and evaluate the role of Generative AI tools in enhancing programming productivity.
2. Apply Python constructs to represent and manipulate data, and design modular code using functions, conditionals, and recursion.
3. Analyze and debug Python programs effectively, including AI-generated code, using appropriate testing and troubleshooting techniques.
4. Compare Python with C, implement basic programs in both languages, and understand core concepts such as memory management and data structures.
5. Identify and optimize performance-critical code segments, and gain foundational insights into data structures and algorithms.

UNIT–I:

Introduction to Programming and GenAI Tools

What is programming?, Role and effectiveness of modern Generative AI tools in programming, Optional: Installing and configuring GenAI tools on your system

UNIT–II:

Data Representation and Manipulation in Python

Fundamental data types and problem-solving mismatches, Objects, expressions, and assignments, Error handling in Python, Using functions to solve sub-problems, Principles of good function design, Introduction to conditionals and recursion

UNIT–III:

Code Comprehension and Debugging in Python

Critiquing code (manual and AI-generated), Testing code and identifying potential errors, Lists and iteration, Asking clarifying questions in problem statements, Debugging using manual and AI-assisted techniques, Introduction to dictionaries

UNIT–IV:

Learning and Comparing Programming Languages (Python vs C)

Key differences between Python and C, Introduction to arrays and structs in C, Memory management basics in C, Implementing a simplified version of Python lists in C

UNIT–V:

Performance-Oriented Programming and Algorithms

Identifying performance bottlenecks in Python code, Translating performance-critical code from Python to C, Introduction to Data Structures and Algorithms (preview)

Textbooks :

1. **Paul Deitel, Harvey Deitel**, *Python for Programmers*, **Pearson Education**, **2nd Edition, 2022**, **ISBN: 978-0136905662**
2. **Brian W. Kernighan, Dennis M. Ritchie**, *The C Programming Language*, **PHI Learning**, **2nd Edition (Indian Reprint, Latest 2021)**, **ISBN: 978-8177581530**
3. **Luciano Ramalho**, *Fluent Python: Clear, Concise, and Effective Programming*, **O'Reilly Media**, **2nd Edition, 2022**, **ISBN: 978-1492056355**
4. **Mark Lutz**, *Learning Python*, **O'Reilly Media**, **5th Edition, Latest Reprint 2023**, **ISBN: 978-9351102014**
5. **Sinan Ozdemir**, *Rust and Python Programming with Generative AI Integration*, **Packt Publishing**, **1st Edition, 2024**, **ISBN: 978-1805124796**

References :

1. Python 3 documentation (https://docs.python.org/3/)
2. The GNU C Reference Manual (https://www.gnu.org/ software/gnu-c-manual/gnu-c-manual.html)
3. Learn AI-Assisted Python Programming (Leo Porter and Daniel Zingaro, Manning Sept 2023, ISBN 9781633437784

	Honor Course	L	T	P	C
		3	0	2	4
ADVANCED COMPUTER VISION					

Course Objectives:

1. Able to apply the core theories and algorithms of computer vision and video processing
2. Understand the state-of-the-art of computer vision and image/video processing,
3. Apply the applications such as vision-based modeling and interaction.

Course Outcomes:

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

1. Identify basic concepts, terminology, theories, models and methods in the field of computer vision.
2. Able to know principles of human visual system.
3. Understanding the advanced methods of computer vision related to GAN, RNN, Deep Dream implementation, LeNet and MNIST etc...
4. Apply a design of a computer vision system for a specific problem.
5. Apply applications of RNN in real time applications.

UNIT – I:

Introduction to Deep Learning, Tensor flow and Keras: What is Deep learning? Why Deep learning, Advantages, and limitations of Deep learning. Tensor flow basics, how to build Deep learning models with Keras and Tensor flow as back end. Tensor board for visualizations.

UNIT - II:

CNN for Vision Tasks: Introduction to CNN, Deep Convolutional networks, LeNet, VGG16Net, and Classification of MNIST hand written digits by CNN and FCNN models.

UNIT - III:

Generative Adversal Networks(GAN's): What is GAN?, DGAN, Some interesting GAN structures, SRGAN, Cycle GAN, info GAN.MNIST using GAN in Tensor flow.

UNIT - IV:

Recurrent Neural Networks: The basic RNN, RNN Cell, RNN variants, RNN topologes, Example applications of RNN. Image captioning and Annotation.

UNIT - V:

Deep Dream and Neural Style Transfer: How the Deep dream algorithm works, Deepdeam implementation in keras and tensor flow. Neural Style Transfer: Content loss, Style loss, Total varianlosses, network training.

Text Books:

1. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systemsby AurélienGéron,Orielly.
2. Deep Learning with Python 1st Edition by François Chollet,MannigPublicatons.
3. Mastering Computer Vision with TensorFlow 2.x: Build advanced computer vision applications using machine learning and deep learning techniquesby Krishnendu Kar ,Packt Publications.
4. Deep Learning with TensorFlow 2 and Keras: Regression, ConvNets, GANs, RNNs,NLP, and more with TensorFlow 2 and the Keras API, 2nd Edition

References:

1. Richard Szeliksy "Computer Vision: Algorithms and Applications" (http://szeliski.org/Book/)
2. Haralick& Shapiro, "Computer and Robot Vision", Vol II
3. G_erardMedioni and Sing Bing Kang "Emerging topics in computer vision"
4. Emanuele Trucco and AllessandroVerri "Introductory Techniques for 3-D Computer Vision", Prentice Hall, 1998.
5. Olivier Faugeras, "Three-Dimensional Computer Vision", The MIT Press, 1993

	Honor Course	L	T	P	C
		3	1	0	4
MATHEMATICAL ESSENTIAL FOR DATA SCIENCE					

Course Objectives:

1. Recall the basics of sets, natural numbers, integers, rational numbers, and real numbers.
2. Learn to use the coordinate system, and plot straight lines.
3. Identify the properties and differences between linear, quadratic, polynomial, exponential, and logarithmic functions.

Course Outcomes:

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

1. Demonstrate understanding of basic mathematical concepts in data science, relating to linear algebra, probability, and calculus.
2. Employ methods related to these concepts in a variety of data science applications.
3. Apply logical thinking to problem-solving in context.
4. Use appropriate technology to aid problem-solving and data analysis.
5. Demonstrate skills in writing mathematics.

UNIT – I:

Set Theory - Number system, Sets and their operations;

Relations and functions - Relations and their types, Functions and their types, Rectangular coordinate system

UNIT – II:

Straight Lines- Slope of a line, Parallel and perpendicular lines, Representations of a Line, General equations of a line, Straight-line fit

Quadratic Functions - Quadratic functions, Minima, maxima, vertex, and slope, Quadratic Equations

UNIT – III:

Algebra of Polynomials - Addition, subtraction, multiplication, and division, Algorithms

UNIT – IV:

Graphs of Polynomials - X-intercepts, multiplicities, end behavior, and turning points, Graphing & polynomial creation

Functions - Horizontal and vertical line tests, Exponential functions, Composite functions, Inverse functions

Logarithmic Functions - Properties, Graphs, Exponential equations, Logarithmic equations

UNIT – V:

Graph Theory - Representation of graphs, Breadth-first search, Depth-first search, Applications of BFS and DFS

Directed Acyclic Graphs - Complexity of BFS and DFS, Topological sorting and longest path, Transitive closure, Matrix multiplication

Graph theory Algorithms - Single source shortest paths, Dijkstra's algorithm, Bellman-Ford algorithm, All-pairs shortest paths, Floyd–Warshall algorithm, Minimum cost spanning trees, Prim's algorithm, Kruskal's algorithm

Text Book:

1. Introductory Algebra: a real-world approach (4th Edition) - by Ignacio Bello

References:

1. Mathematical Foundations Of Data Science Using Rby Emmert-Streib Frank.

	Honor Course	L	T	P	C
		3	1	0	4
INTRODUCTION TO DATA SCIENCE					

Course Objectives:

1. The course teaches critical concepts and skills in computer programming and statistical inference, in conjunction with hands-on analysis of real-world datasets, including economic data, document collections, geographical data, and social networks.
2. It delves into social issues surrounding data analysis such as privacy and design.
3. Explain out-of-sample evaluation metrics

Course Outcomes:

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

1. Apply dimensionality reduction tools such as principle component analysis
2. Evaluate outcomes and make decisions based on data
3. Understand how to Use exploratory tools such as clustering and visualization tools to analyze data.
4. Apply dimensionality reduction tools such as principle component analysis
5. Able to know how to perform basic analysis of network data.

UNIT – I:

Introduction

Introduction to Data Science – Evolution of Data Science – Data Science Roles – Stages in a Data Science Project – Applications of Data Science in various fields – Data Security Issues.

UNIT – II:

Data Collection and Data Pre-Processing

Data Collection Strategies – Data Pre-Processing Overview – Data Cleaning – Data Integration and Transformation – Data Reduction – Data Discretization.

UNIT – III:

Exploratory Data Analytics

Descriptive Statistics – Mean Standard Deviation, Skewness and Kurtosis – Box Plots – Pivot Table – Heat Map Correlation Statistics – ANOVA.

UNIT – IV:

Model Development

Simple and Multiple Regression – Model Evaluation using Visualization – Residual Plot – Distribution Plot – Polynomial Regression and Pipelines – Measures for In-sample Evaluation – Prediction and Decision Making.

UNIT – V:

Model Evaluation

Generalization Error – Out-of-Sample Evaluation Metrics – Cross Validation – Over fitting –Under Fitting and Model Selection – Prediction by using Ridge Regression – Testing Multiple Parameters by using Grid Search.

Text Books:

1. Data Science for Beginners, by Andrew Park
2. The Art of Data Science — A Guide for Anyone Who Works With Data, by Roger D. Peng and Elizabeth Matsui.

References:

1. JojoMoolayil, "Smarter Decisions : The Intersection of IoT and Data Science",PACKT, 2016.
2. Cathy O'Neil and Rachel Schutt , "Doing Data Science", O'Reilly, 2015.
3. David Dietrich, Barry Heller, Beibei Yang, "Data Science and Big data Analytics",EMC 2013
4. Raj, Pethuru, "Handbook of Research on Cloud Infrastructures for Big DataAnalytics", IGI Global.

	Honor Course	L	T	P	C

		3	1	0	4
DATA ANALYTICS AND VISUALIZATION					

Course Objectives:

1. To demonstrate expert knowledge of data analysis, statistics, tools, techniques and technologies of data analytics and Visualization.
2. To enable learners to develop knowledge and skills in current and emerging areas of data analytics and Visualization.
3. To formulate and implement a novel research idea and conduct research in the field of data analytics and Visualization.

Course Outcomes:

After completing the course, student will be able to:

1. Present data with visual representations for your target audience, task, and data;
2. Identify appropriate data visualization techniques given particular requirements imposed by the data;
3. Display types, Geospatial displays, Interactivity
4. Data Definitions and Analysis Techniques
5. Implement the analytic algorithms and Basic analysis techniques

UNIT -1:

Introduction and Tableau Primer: Introduction to data visualization Data for data graphics Tableau introduction

UNIT-2:

Design Principles: Design principles Categorical, time series, and statistical data graphics

UNIT-3:

Display types, Geo spatial displays, Interactivity: Storytelling Multivariate displays, Geospatial displays, Dashboards, interactive and animated displays

UNIT-4:

Data Definitions and Analysis Techniques: Elements, Variables, and Data categorization, Levels of Measurement, Data management and indexing, Introduction to statistical learning.

Descriptive Statistics: Measures of central tendency, Measures of location of dispersions.

UNIT-5:

Basic analysis techniques: Statistical hypothesis generation and testing, Chi-Square test,t-Test,Analysis of variance, Correlation analysis, Maximum likelihood test.

Text Books:

1. Sosulski, K. (2018). Data Visualization Made Simple: Insights into BecomingVisual. New York: Routledge.

2. Probability & Statistics for Engineers & Scientists (9th Edn.), Ronald E. Walpole, Raymond H. Myers, Sharon L. Myers and Keying Ye, Prentice Hall Inc.
3. The Elements of Statistical Learning, Data Mining, Inference, and Prediction (2nd Edn.), Trevor Hastie Robert Tibshirani Jerome Friedman, Springer, 2014

References:

1. An Introduction to Statistical Learning: with Applications in R, G James, D. Witten, T Hastie, and R. Tibshirani, Springer, 2013
2. Software for Data Analysis: Programming with R (Statistics and Computing), John M. Chambers, Springer
3. Mining Massive Data Sets, A. Rajaraman and J. Ullman, Cambridge University Press, 2012
4. Advances in Complex Data Modeling and Computational Methods in Statistics, Anna Maria Paganoni and Piercesare Secchi, Springer, 2013

Optional readings:

1. Few, S. (2012). Show me the numbers: Designing tables and graphs toenlighten. Burlingame, CA: Analytics Press.
2. Few, S. (2006). Information dashboard design: The effective visualcommunication of data. Sebastopol: O'Reilly.
3. Ware, C & Kaufman, M. (2008). Visual thinking for design. Burlington: MorganKaufmann Publishers.
4. Wong, D. (2011). The Wall Street Journal guide to information graphics: The dosand don'ts of presenting data, facts and figures. New York: W.W. Norton& Company.
5. Yau, N. (2011). Visualize This: The FlowingData Guide to Design, Visualization,and Statistics. Indianapolis: O'Reilly.
6. Yau, N. (2013). Data Points: Visualization that means something. Indianapolis: O'Reilly.

	Honor Course	L	T	P	C
		3	1	0	4
GRAPH ANALYTICS					

Course Objectives:

1. To provide a solid foundation in graph theory concepts and graph representations.
2. To familiarize students with classical graph algorithms and their applications in real-world problems.
3. To introduce advanced topics such as community detection, centrality measures, and network analysis techniques.
4. To explore graph-based machine learning models, including node embeddings and Graph Neural Networks (GNNs).
5. To equip students with hands-on skills using modern graph analytics tools and frameworks like Python, Neo4j, and TigerGraph.
6. To enable analysis of large-scale and complex graph data for applications in diverse domains such as social networks, fraud detection, and knowledge graphs.

Course Outcomes:

1. Demonstrate knowledge of graph representations and core graph properties.
2. Implement classical graph algorithms (traversals, shortest paths, centrality).
3. Apply community detection and clustering methods to real-world graphs.
4. Use node embeddings and graph neural networks for analytics tasks.
5. Analyze graphs using Python/Neo4j frameworks and handle large-scale graph data.

UNIT I:

Introduction to Graph Theory & Representations- Types of graphs (directed, undirected, weighted), Graph representations (adjacency matrix, list, edge list), Graph properties & terminology, Graph operations and transformations

UNIT II:

Graph Algorithms & Centrality Measures- Breadth-First Search (BFS), Depth-First Search (DFS), Shortest path algorithms (Dijkstra, Bellman-Ford), Centrality measures (Degree, Betweenness, Closeness, Eigenvector centrality, PageRank), Applications to social networks and recommendation systems

UNIT III:

Community Detection & Network Analysis - Community detection methods (Louvain, Girvan-Newman), Modularity, Network motifs, Structural balance theory, Triadic closure, Subgraph analysis, Case-study applications on social and biological networks

UNIT IV:

Graph-based Machine Learning & Embeddings- Graph-based ML: Graph Convolutional Networks (GCNs), Graph Neural Networks (GNNs), Node embedding methods (node2vec, DeepWalk), Graph kernels, Link prediction, Node classification

UNIT V:

Large-Scale & Parallel Graph Analytics - Random graph models: Erdos–Rényi, Small-world, Scale-free; Spectral graph analysis, GraphBLAS and linear-algebraic graph computation, Parallel graph analytics frameworks (GraphX/Pregel), scalability considerations

UNIT VI:

Applications, Tools & Case Studies- Graph DBs (Neo4j, Cypher queries), Graph analytics pipelines, Applications: fraud detection, social media mining, citation networks, knowledge graphs; Integration into data science workflows using Python, Neo4j, TigerGraph

Text books :

1. **Timothy Eastridge –** *Graph Data Science with Python and Neo4j*, **Enterprise Strategies**, 2024
2. **Tomaž Bratanic –** *Graph Algorithms for Data Science* (with Neo4j examples), **2024** (self-published)

Reference Books :

1. **Victor Lee, Phuc Nguyen & Alexander Thomas –** *Graph-Powered Analytics and Machine Learning with TigerGraph*, **O'Reilly Media**, 2023
2. **Bonia diestel et al. –** *Graph Theory and Applications*, **Springer**, 2008 (classic with strong fundamentals)
3. **Albert-László Barabási –** *Network Science*, **Cambridge Univ. Press**, 2016
4. **M. E. J. Newman –** *Networks: An Introduction*, **Oxford University Press**, 2010

JNTUGV- COLLEGE OF ENGINEERING VIZIANAGARAM

DEPARTMENT OF INFORMATION TECHNOLOGY

B.TECH- IT (R23)-MINORS

MINOR DEGREE IN INFORMATION TECHNOLOGY

(For Non-CSE / IT Students)

(I) Artificial Intelligence

S.No	Subject Title	L	T	P	C
1	Introduction to Artificial Intelligence	3	0	0	3
2	Mathematics for Machine Learning	3	0	0	3
3	Machine Learning	3	0	0	3
4	Deep Learning	3	0	0	3
5	03 MOOCS courses @ 2credits each (8-week course) 02 MOOCS courses @ 3credits each (12-week course) (Any CSE/IT related **Program Core subject** from NPTEL/ SWAYAM course other than the courses listed above needs to be taken)				6
				Total	**18**

(II) Computer Security

S.No	Subject Title	L	T	P	C
1	Cyber Security	3	0	0	3
2	Cyber Crime Investigation and Digital Forensics	3	0	0	3
3	Cryptography and Applications	3	0	0	3
4	Blockchain Technology	3	0	0	3
5	03 MOOCS courses @ 2credits each (8-week course) 02 MOOCS courses @ 3credits each (12-week course) (Any CSE/IT related **Program Core subject** from NPTEL/SWAYAM course other than the courses listed above needs to be taken)				6
				Total	**18**

(III) Programming and Web Development

S.No	Subject Title	L	T	P	C
1	Object Oriented Programming through Java	2	0	2	3
2	Python Programming	2	0	2	3
3	Basic Web Designing	2	0	2	3
4	Advanced Web Technologies	2	0	2	3
5	03 MOOCS courses @ 2credits each (8-week course) 02 MOOCS courses @ 3credits each (12-week course) (Any CSE/IT related **Program Core subject** from NPTEL/SWAYAM course other than the courses listed above needs to be taken)				6
				Total	**18**

(IV) Advanced Computing

S.No	Subject Title	L	T	P	C
1	Computer Organization and Architecture	3	0	0	3
2	Distributed Systems	3	0	0	3
3	Cloud Computing	3	0	0	3
4	Quantum Computing	3	0	0	3
5	03 MOOCS courses @ 2credits each (8-week course) 02 MOOCS courses @ 3credits each (12-week course) (Any CSE/IT related **Program Core subject** from NPTEL/SWAYAM course other than the courses listed above needs to be taken)				6
				Total	**18**

Note: Students who have registered for minors program can opt a maximum of two (02) courses per semester.

	Minor Course	L	T	P	C
		3	1	0	4
INTRODUCTION TO ARTIFICIAL INTELLIGENCE					

Course Objectives:

1. To have a basic proficiency in a traditional AI language including an ability to write simple to intermediate programs and an ability to understand code written in that language
2. To understand the basic issues of knowledge representation and blind and heuristic search, as well as an understanding of other topics such as minimax, resolution, etc. that play an important role in AI programs
3. To have a basic understanding of some of the more advanced topics of AI such as learning, natural language processing, agents and robotics, expert systems, and planning.

Course Outcomes:

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

1. Outline problems that are amenable to solution by AI methods, and which AI methods may be suited to solving a given problem
2. Apply the language/framework of different AI methods for a given problem
3. Implement basic AI algorithms- standard search algorithms or dynamic programming
4. Design and carry out an empirical evaluation of different algorithms on problem formalization, and state the conclusions that the evaluation supports
5. Design Expert Systems using fuzzy logic theory

UNIT- I:

Introduction: history, intelligent systems, foundations of AI, applications, tic-tac-toe game playing, development of AI languages, current trends.

UNIT -II:

Problem Solving: state-space search and control strategies: Introduction, general problem solving, characteristics of problem, exhaustive searches, heuristic search techniques, iterative deepening A*, constraint satisfaction.

Problem reduction and game playing: Introduction, problem reduction, game playing, alpha beta pruning, two-player perfect information games.

UNIT –III:

Logic concepts: Introduction, propositional calculus, proportional logic, natural deduction system, axiomatic system, semantic tableau system in proportional logic, resolution refutation in proportional logic, predicate logic.

UNIT -IV:

Knowledge representation: Introduction, approaches to knowledge representation, knowledge representation using semantic network, extended semantic networks for KR, knowledge representation using frames.

Advanced knowledge representation techniques: Introduction, conceptual dependency theory, script structure, CYC theory, case grammars, semantic web

UNIT–V:

Expert system and applications: Introduction phases in building expert systems, expert system versus traditional systems Uncertainty measure: probability theory: Introduction, probability theory, Bayesian belief networks, certainty factor theory, dempster-shafer theory .

Fuzzy sets and fuzzy logic: Introduction, fuzzy sets, fuzzy set operations, types of membership functions, multi valued logic, fuzzy logic, linguistic variables and hedges, fuzzy propositions, inference rules for fuzzy propositions, fuzzy systems.

Text Books:

1. Artificial Intelligence- Saroj Kaushik, CENGAGE Learning
2. Artificial intelligence, A modern Approach , 2nded, Stuart Russel, Peter Norvig,PEA

References:

1. Artificial Intelligence- Deepak Khemani, TMH,2013
2. Introduction to Artificial Intelligence, Patterson,PHI
3. Atificial intelligence, structures and Strategies for Complex problem solving,-George F Lugar, 5thed, PEA

E-Resources:

1. https://nptel.ac.in/courses/106/105/106105077/
2. http://aima.cs.berkeley.edu/

	Minor Course	L	T	P	C
		3	1	0	4
MATHEMATICS FOR MACHINE LEARNING					

Course Objectives:

1. The purpose of this course is to provide a mathematically rigorous introduction to these developments with emphasis on methods and their analysis.
2. Explain and apply matrix decomposition techniques
3. Explain parameter estimation using the Maximum Likelihood method

Course Outcomes:

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

1. Understand the strengths and weaknesses of many popular machine learning approaches.
2. Justify the underlying mathematical relationships within and across Machine Learning algorithms.
3. Evaluate the several areas of mathematics beyond calculus
4. Solve problems in a range of mathematical applications
5. Apply various methods to compute the probabilities of events, Analyze and interpret statistical data using appropriate probability distributions.

UNIT-I:

Linear Algebra: Systems of Linear Equations, Matrices, Solving systems of linear equations, Vector Spaces, Linear Independence, Basis and Rank, Linear Mappings.

Analytic Geometry: Norms, Inner Products, Lengths and Distances, Angles and Orthogonality, Orthonormal Basis, Orthogonal Complement, Inner Product of Functions, Orthogonal Projections.

UNIT-II:

Matrix Decompositions: Determinant and Trace, Eigen values and Eigen vectors, Cholesky Decomposition, Eigen decomposition and Diagonalization, Singular Value Decomposition, Matrix Approximation.

Vector Calculus: Differentiation of Univariate Functions, Partial differentiation and Gradients, Gradients of vector valued functions, Gradients of Matrices, Useful identities for computing gradients, Back propagation and Automatic Differentiation

UNIT-III:

Probability and Distributions: Construction of a Probability space, Discrete and Continuous probabilities, sum rule, product rule and Bayes Theorem, Summary statistics and Independence, Gaussian Distribution.

Continuous Optimization: Optimization using Gradient Descent, Constrained optimization and Lagrange Multipliers, Convex Optimization.

UNIT-IV:

Linear Regression: Problem Formulation, Parameter Estimation, Bayesian Linear Regression, Maximum Likelihood as Orthogonal Projection.

Dimensionality Reduction with Principal Component Analysis: Problem setting, Maximum Variance Perspective, Projection Perspective, Eigenvector computation and Low Rank Approximations, PCA in High Dimensions, Latent Variable Perspective.

UNIT-V:

Density Estimation with Gaussian Mixture Models: Gaussian Mixture Model, Parameter Learning via Maximum Likelihood, EM Algorithm, Latent-Variable Perspective.

Classification with Support Vector Machines: Separating Hyper planes, Primal Support Vector Machine, Dual Support Vector Machine, Kernels, Numerical Solution.

Text Books:

1. https://mml-book.github.io/book/mml-book.pdf - c 2021 M. P. Deisenroth, A. A. Faisal, C. S. Ong. Published by Cambridge University Press (2020).

References:

1. https://www.youtube.com/watch?v=1VSZtNYMntM

	Minor Course	L	T	P	C
		3	1	0	4
MACHINE LEARNING					

Course Objectives:

1. To learn well -known machine learning algorithms
2. To evaluate and compare the performance of various machine learning algorithms
3. Able to differentiate regression models and distance based models and ANNS.

Course Outcomes:

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

1. Recognize the characteristics of machine learning algorithms and their applications to real world problems
2. Able to differentiate linear and logistic regressions.
3. Able to write and evaluate hypothesis
4. Understand the concepts of Artificial neural networks
5. Can apply kernel methods to solve real world problems.

UNIT I:

Introduction: Well -posed learning problems, designing a learning system, Perspectives, and issues in machine learning. Concept learning and the general to specific ordering – Introduction, A concept learning task, Concept learning as search, Find -S: finding a maximally specific hypothesis, Version spaces and the candidate elimination algorithm, Remarks on version spaces and candidate elimination, Inductive bias.

UNIT II:

Linear Regression & Logistic Regression- Predicting Numeric values : Regression - Finding the best fit lines with linear regression, locally weighted linear regression, Shrinking Coefficients, The bias / Variance tradeoff.

Logistic Regression: Classification with logistic regression and the sigmoid function, Using optimization to find the best regression coefficients.

UNIT III:

Artificial Neural Networks : Introduction, Neural network representation, Appropriate problems for neural network learning, Perceptions, Multilayer networks and the back propagation algorithm, Remarks on the back propagation algorithm, An illustrative example face recognition, Advanced topics in artificial neural networks

UNIT IV:

Evaluation Hypotheses: Motivation, Estimation hypothesis accuracy, Basics of sampling theory, A general approach for deriving confidence intervals, Difference in error of two

hypotheses, Comparing learning algorithms.

UNIT V:

Support Vector Machines: Separating data with the maximum margin, finding the maximum margin, efficient optimization with SMO algorithm, speeding up optimization with full platt SMO, Using Kernels for more Complex data.

Text Books:

1. Machine Learning ,Tom M. Mitchell, MGH
2. Machine Learning in Action, Peter Harington, 2012, Cengage.

References:

1. Introduction to Machine Learning, EthemAlpaydin, PHI, 2004
2. A course in Machine Learning , Hall Daum'e III

	Minor Course	L	T	P	C
		3	1	0	4
DEEP LEARNING					

Course Objectives:

At the end of the course, the students will be expected to:

1. Learn deep learning methods for working with sequential data,
2. Learn deep recurrent and memory networks,
3. Learn deep Turing machines

Course Outcomes:

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

1. Demonstrate the basic concepts fundamental learning techniques and layers.
2. Discuss the Neural Network training, various random models.
3. Explain different types of deep learning network models.
4. Classify the Probabilistic Neural Networks and Sequence model neural networks.
5. Implement tools on Deep Learning techniques.

UNIT I:

Introduction: Various paradigms of learning problems, Perspectives and Issues in deep learning framework, review of fundamental learning techniques.

Feed forward neural network: Artificial Neural Network, activation function, multi-layer neural network.

UNIT II:

Training Neural Network: Risk minimization, loss function, back propagation, regularization, model selection, and optimization.

Conditional Random Fields: Linear chain, partition function, Markov network, Belief propagation, Training CRFs, Hidden Markov Model, Entropy.

UNIT III:

Deep Learning: Deep Feed Forward network, regularizations, training deep models, dropouts, Convolution Neural Network, Recurrent Neural Network, and Deep Belief Network.

UNIT IV:

Probabilistic Neural Network: Hopfield Net, Boltzmann machine, RBMs, Sigmoid net, Auto encoders.

Sequence Modeling: LSTM, Gated RNNs & Deep Generative Models

UNIT V:

Applications: Object recognition, sparse coding, computer vision, natural language processing.

Introduction to Deep Learning Tools: Caffe, Theano, Torch.

Text Books:

1. Good fellow, I., Bengio,Y., and Courville, A., Deep Learning, MIT Press, 2016..
2. Bishop, C., M., Pattern Recognition and Machine Learning, Springer, 2006.

References:

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

	Minor Course	L	T	P	C
		3	1	0	4
CYBER SECURITY					

Course Objectives:

In this course, the student will learn about

1. The essential building blocks and basic concepts around cyber security such as Confidentiality, Integrity, Availability, Authentication, Authorization, Vulnerability, Threat & Risk and so on.
2. Analyze various browser-based attacks
3. Explore strategic network defense mechanisms

Course Outcomes:

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

1. Illustrate the broad set of technical, social & political aspects of Cyber Security and security management methods to maintain security protection
2. Appreciate the vulnerabilities and threats posed by criminals, terrorist and nation states to national infrastructure
3. Illustrate the nature of secure software development and operating systems
4. Demonstrate the role security management plays in cyber security defense and legal and social issues at play in developing solutions
5. Assess privacy concerns in data mining and web-based applications

UNIT –I:

Introduction: Introduction to Computer Security, Threats, Harm, Vulnerabilities, Controls, Authentication, AccessControl, and Cryptography, Authentication, Access Control, Cryptography.

Programs and Programming: Unintentional (Non-malicious) Programming Oversights, Malicious Code—Malware, Countermeasures.

UNIT –II:

Web Security: User Side, Browser Attacks, Web Attacks Targeting Users, Obtaining User or Website Data, Email Attacks.

Operating Systems Security: Security in Operating Systems, Security in the Design of Operating Systems, Root kit.

UNIT -III:

Network Security: Network Concepts, Threats to Network Communications, Wireless Network Security, Denial of Service, Distributed Denial-of-Service Strategic Defenses: Security Countermeasures, Cryptography in Network Security, Firewalls, Intrusion Detection and Prevention Systems, Network Management .

Cloud Computing and Security: Cloud Computing Concepts, Moving to the Cloud, Cloud

Security Tools and Techniques, Cloud Identity Management, Securing IaaS.

UNIT- IV:

Privacy: Privacy Concepts, Privacy Principles and Policies, Authentication and Privacy, Data Mining, Privacy on the Web, Email Security, Privacy Impacts of Emerging Technologies, Where the Field Is Headed.

Management and Incidents: Security Planning, Business Continuity Planning, Handling Incidents, Risk Analysis, Dealing with Disaster.

UNIT –V:

Legal Issues and Ethics: Protecting Programs and Data, Information and the Law, Rights of Employees and Employers, Redress for Software Failures, Computer Crime, Ethical Issues in Computer Security, Incident Analysis with Ethics Emerging Topics: The Internet of Things, Economics, Computerized Elections, Cyber Warfare.

Text Books:

1. Pfleeger, C.P., Security in Computing, Prentice Hall, 2010, 5th edition.
2. Schneier, Bruce. Applied Cryptography, Second Edition, John Wiley & Sons, 1996

References:

1. Rhodes-Ousley, Mark. Information Security: The Complete Reference, Second Edition, Information Security Management: Concepts and Practice, McGraw-Hill, 2013.
2. Whitman, Michael E. and Herbert J. Mattord. Roadmap to Information Security for IT and Infosec Managers. Boston, MA: Course Technology, 2011.

	Minor Course	L	T	P	C
		3	1	0	4
CYBER CRIME INVESTIGATION AND DIGITAL FORENSICS					

Course Objectives:

1. Able to identify security risks and take preventive steps
2. To understand the forensics fundamentals.
3. To understand the evidence capturing process.

Course Outcomes:

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

1. Acquire the definition of computer forensics fundamentals.
2. Describe the types of computer forensics technology
3. Analyze various computer forensics systems.
4. Illustrate the methods for data recovery, evidence collection and data seizure.
5. Summarize duplication and preservation of digital evidence.

UNIT–I:

Introduction: Introduction and Overview of Cyber Crime, Nature and Scope of Cyber Crime, Types of Cyber Crime: Social Engineering, Categories of Cyber Crime, Property Cyber Crime.

UNIT–II:

Cyber Crime Issues: Unauthorized Access to Computers, Computer Intrusions, White collar Crimes, Viruses and Malicious Code, Internet Hacking and Cracking, Virus Attacks, Pornography, Software Piracy, Intellectual Property, Mail Bombs, Exploitation ,Stalking and Obscenity in Internet, Digital laws and legislation, Law Enforcement Roles and Responses.

UNIT–III:

Investigation: Introduction to Cyber Crime Investigation, Investigation Tools, e-Discovery, 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:

Digital Forensics: Introduction to Digital Forensics, Forensic Software and Hardware, Analysis and Advanced Tools, Forensic Technology and Practices, Forensic Ballistics and Photography, Face, Iris and Fingerprint Recognition, Audio Video Analysis, Windows System Forensics, Linux System Forensics, Network Forensics.

UNIT– V:

Laws And Acts: Laws and Ethics, Digital Evidence Controls, Evidence Handling Procedures, Basics of Indian Evidence ACT IPC and CrPC ,Electronic Communication Privacy ACT, Legal Policies.

Text Books:

1. Nelson, B., Phillips, A., & Steuart, C., *Guide to Computer Forensics and Investigations*, Cengage Learning, 2018, 6th Edition.
2. Casey, Eoghan, *Digital Evidence and Computer Crime: Forensic Science, Computers and the Internet*, Academic Press, 2011, 3rd Edition.
3. Brenner, Susan W., *Cybercrime: Criminal Threats from Cyberspace*, Praeger Security International, 2010.

	Minor Course	L	T	P	C
		3	1	0	4
CRYPTOGRAPHY AND APPLICATIONS					

Course Objective:

This course aims at training students to master the:

1. The concepts of classical encryption techniques and concepts of finite fields and number theory
2. Working principles and utilities of various cryptographic algorithms including secret key cryptography, hashes and message digests, and public key algorithms.
3. Design issues and working principles of various authentication protocols and PKI standards.

Course Outcomes:

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

1. Identify information security goals and acquire fundamental knowledge on the concepts of finite fields and number theory
2. Compare and apply different encryption and decryption techniques to solve problems related to confidentiality and authentication
3. Apply the knowledge of cryptographic checksums and evaluate the performance of different message digest algorithms for verifying the integrity of varying message sizes.
4. Apply different digital signature algorithms to achieve authentication and create secure applications
5. Apply network security basics, analyze different attacks on networks and evaluate the performance of security protocols like SSL, IPSec, and PGP

UNIT- I:

Introduction to Security: Security Attacks, Security Services, Security Mechanisms, Fundamental Security Design Principles, Attack Surfaces and Attack Trees, a Model for Network Security

Mathematics of Cryptography: Algebraic Structures (Groups, Rings, Fields and Galois Fields), Divisibility and the Division Algorithm, The Euclidean Algorithm, Modular Arithmetic, Prime Numbers, Fermat's and Euler's Theorems, Testing for Primality, The Chinese Remainder Theorem, Discrete Logarithms

UNIT- II:

Classical Encryption Techniques: Symmetric Cipher Model, Substitution Techniques, Transposition Techniques, Rotor Machines, Steganography

Block Ciphers: Traditional Block Cipher Structure, The Data Encryption Standard, The

Strength of DES, Block Cipher Design Principles, Advanced Encryption Standard, AES Structure, AES Transformation Functions, AES Key Expansion, Multiple Encryption and Triple DES, Block Cipher Modes of Operation

UNIT- III:

Public-Key Cryptography: Principles of Public-Key Cryptosystems, The RSA Algorithm, Diffie- Hellman Key Exchange, Elgamal Cryptographic System, Elliptic Curve Cryptography, **Cryptographic Hash Functions**: Applications of Cryptographic Hash Functions, Requirements and Security, Secure Hash Algorithm (SHA), **Message Authentication Codes**: Requirements for Message Authentication Codes, HMAC, CMAC

UNIT- IV:

Digital Signatures: Digital Signatures, Elgamal Digital Signature Scheme, Schnorr Digital Signature Scheme, NIST Digital Signature Algorithm, Elliptic Curve Digital Signature Algorithm

Key Management and Distribution: Symmetric Key Distribution Using Symmetric Encryption, Symmetric Key Distribution Using Asymmetric Encryption, Distribution of Public Keys, X.509 Certificates, Public-Key Infrastructure

User Authentication: Remote User-Authentication Principles, Remote User-Authentication Using Symmetric Encryption, Kerberos, Remote User-Authentication Using Asymmetric Encryption:

UNIT -V:

Transport-Level Security: Web Security Considerations, Transport Layer Security, Secure Shell (SSH)

Electronic Mail Security: S/MIME, Pretty Good Privacy

IP Security: IP Security Overview, Encapsulating Security Payload, Combining Security Associations, Internet Key Exchange

Text Book:

1. Cryptography and Network Security, William Stallings, 8th Edition, Pearson Education

References:

1. Cryptography, Network Security and Cyber Laws, Bernard L. Menezes, Ravinder Kumar, Cengage Learning.
2. Cryptography and Network Security, Behrouz A Forouzan, DebdeepMukhopadhyaya, 3rd Edition, Mc-GrawHill.
3. Network Security Illustrated, Jason Albanese, Wes Sonnenreich, and McGraw Hill.

	Minor Course	L	T	P	C
		3	1	0	4
BLACK CHAIN TECHNOLOGY					

Course Objectives:

1. To provide conceptual understanding of the function of Block chain as a method of securing distributed ledgers.
2. To understand the structure of a Block chain and why/when it is better than a simple distributed database
3. To make students understand the technological underpinnings of Block chain operations as distributed data structures and decision making systems

Course Outcomes:

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

1. Define and explain the fundamentals of Block chain.
2. Understand decentralization and the role of Block chain in it.
3. Understand and analyze Bitcioin Crypto currency and underlying Block chain network.
4. Understand Etherium currency and platform, and develop applications using Solidity.
5. Understand Hyper ledger project and its components; critically analyze the challenges and future opportunities in Block chain technology.

UNIT-I:

Introduction: History and basics, Types of Block chain, Consensus, CAP Theorem. Cryptographic Hash Functions: Properties of hash functions, Secure Hash Algorithm, Merkle trees, Patricia trees.

UNIT-II:

Decentralization: Decentralization using Block chain, Methods of decentralization, decentralization framework, Blockchain and full ecosystem decentralization, Smart contracts, Decentralized Organizations, Platforms for decentralization.

UNIT-III:

Bitcoin: Introduction to Bitcoin, Digital keys and addresses, Transactions, Blockchain, The Bitcoin network, Bitcoin payments, Bitcoin Clients and APIs, Alternatives to Proof of Work, Bitcoin limitations.

UNIT-IV:

Etherium: Smart Contracts, Introduction to Ethereum, The Ethereum network, Components of the Ethereum ecosystem, Blocks and Blockchain, Fee schedule, Ethereum Development

Environment, Solidity.

UNIT-V:

Hyperledger: Introduction, Hyperledger Projects, Protocol, Architecture, Hyperledger Fabric, Sawtooth Lake, Corda.

Challenges and Opportunities: Scalability, Privacy, Blockchain for IoT, Emerging trends

Text Book:

1. Mastering Blockchain, Imran Bashir, Second Edition, Packt Publishing.

References:

1. Mastering Bitcoin: Unlocking Digital Cryptocurrencies, Andrea Antonopoulos, and O'Reilly.
2. Blockchain Blueprint for a New Economy, Melanie Swan, O'Reilly.
3. Mastering Bitcoin: Programming the Open Blockchain, Antonopoulos, Andreas M. O'Reilly.
4. Blockchain Technology: Cryptocurrency and Applications, S. Shukla, M. Dhawan, S. Sharma, S. Venkatesan, Oxford University Press.

	Minor Course	L	T	P	C
		3	0	2	4
OBJECT ORIENTED PROGRAMMING THROUGH JAVA					

Course Objectives:

1. Implementing programs for user interface and application development using core java principles.
2. Focus on object oriented concepts and java program structure and its installation.
3. Comprehension of java programming constructs, control structures in Java Programming Constructs.

Course Outcomes:

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

1. Understand Java programming concepts and utilize Java Graphical User Interface in Program writing.
2. Write, compile, execute and troubleshoot Java programming for networking concepts.
3. Build Java Application for distributed environment.
4. Design and Develop multi-tier applications.
5. Identify and Analyze Enterprise applications.

UNIT I:

Introduction to OOP - Introduction, Need of Object Oriented Programming, Principles of Object Oriented Languages, Procedural languages Vs OOP, Applications of OOP, History of JAVA, Java Virtual Machine, Java Features, Program Structures, Installation of JDK1.6.

UNIT II:

Variables, Primitive Data types, Identifiers- Naming Conventions, Keywords, Literals, Operators-Binary, Unary and Ternary, Expressions, Precedence rules and Associativity, Primitive Type Conversion and Casting, Flow of Control-Branching, Conditional Loops.

Classes and Objects- Classes, Objects, Creating Objects, Methods, Constructors-Constructor Overloading, Cleaning up Unused Objects-Garbage Collector, Class Variable and Methods-Static Keyword, this keyword, Arrays, Command Line Arguments.

UNIT III:

Inheritance: Types of Inheritance, Deriving Classes using Extends Keyword, Method Overloading, Super Keyword, Final Keyword, Abstract Class.

Interfaces, Packages and Enumeration: Interface-Extending Interface, Interface Vs Abstract Classes, Packages-Creating Packages, Using Packages, Access Protection, java.lang Package.

Exceptions & Assertions - Introduction, Exception Handling Techniques-try...catch, throw, throws, finally block, User Defined Exception, Exception Encapsulation and Enrichment, Assertions.

UNIT IV:

Multi-Threading: java.lang.Thread, The main Thread, Creation of New Threads, Thread Priority, Multithreading- Using isAlive() and join(), Synchronization, Suspending and Resuming Threads, Communication between Threads.

Input/Output: Reading and Writing data, java.io package.

Applet: Applet Class, Applet Structure, Applet Life Cycle, Sample Applet Programs.

UNIT V:

Event Handling: Event Delegation Model, Sources of Event, Event Listeners, Adapter Classes, Inner Classes.

Abstract Window Toolkit : Importance of AWT, Java.awt.package, Components and Containers, Button, Label, Check Box, Radio Buttons, List Boxes, Choice Boxes, Text Field and Text Area, Container Classes, LayOuts, Menu, Scroll bar.

Swings: Introduction, JFrame, JApplet, JPanel, Components in Swings, Layout Managers, List and JScroll Pane, SplitPane, JTabbedPane, JTree, DialogBox, Pluggable Look and Feel.

Text Books:

1. The Complete Reference Java, 8ed, Herbert Schildt, TMH.
2. Programming in JAVA, Sachin Malhotra, Saurabh Choudhary, Oxford.
3. JAVA for Beginners, 4e, Joyce Farrell, Ankit R. Bhavsar, Cengage Learning.

References:

1. JAVA Programming, K.Rajkumar, Pearson.
2. Core JAVA, Black Book, Nageswara Rao, Wiley, Dream Tech.
3. Core JAVA for Beginners, Rashmi Kanta Das, Vikas.
4. Object Oriented Programming through JAVA , P Radha Krishna , University Press.
5. Object oriented programming with JAVA, Essentials and Applications, Raj Kumar Bhuyya,Selvi, Chu TMH.
6. Introduction to Java Programming, 7th ed, Y Daniel Liang, Pearson.

	Minor Course	L	T	P	C
		3	0	2	4
PYTHON PROGRAMMING					

Course Objectives:

1. Introduction to Scripting Language
2. Exposure to various problems solving approaches of computer science
3. **Teach the use of multiple except blocks** to handle different types of exceptions individually and appropriately.

Course Outcomes:

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

1. Understand the fundamentals of scripting language and its learning environment.
2. Acquire the knowledge of data types, operators and control structures.
3. Understand Object oriented concepts and apply the concepts of data structures to real world data.
4. Apply the concept of modularity and implement different packages to solve complex problems. Understand Object oriented concepts and handle different errors through exceptions.
5. Develop multithreaded application using standard libraries.

UNIT-I:

Features and History of Python, Print and Input functions, variables, keywords, comments, **Types:** Numerical Types (int, float, complex), Strings, Boolean, Type Conversion, **Operators:** Arithmetic, Relational, Logical, Bitwise, Assignment, Identity, Membership, **Control Flow:** Indentation, if-elseif-else, while, for, break, continue, pass, else-with loops

UNIT-II:

Functions: Introduction, Required Arguments, Default Arguments, Keyword Arguments, Variable Number of Arguments, Variable Scope and Lifetime, global variables, Lambda Functions, Command Line Arguments

Object Oriented Programming: Classes and Objects, built-in class methods and attributes, 'self', constructor, destructor, inheritance, data hiding, overriding methods and overloading operators

UNIT-III:

Data Structures: Lists, Nested Lists, List Comprehensions, Tuples and Sequences, Sets, Dictionaries

File I/O: opening, closing, reading and writing.

UNIT-IV:

Exception Handling: Exceptions, Multiple Except Blocks, Multiple Exceptions in a Single Block, Except Block without Exception, the else Clause, Raising Exceptions, Built-in and User-defined Exceptions, The finally block, Introduction modules, import and from-import, Packages in Python, used defined modules and packages, PIP.

UNIT-V:

The Python Standard Library: numeric and mathematical modules, string processing, date & time, calendar, operating system, web browser

Graphics with turtle: Motion Control, Pen, Colour, Fill, multiple turtles, reset and clear

GUI design with tkinter: Button, Canvas, Check button, Entry, Frame, Label, Listbox, Menu, Menu button, Message, Radio button, Scale, Scrollbar, Text

Text Books:

1. Python Programming using problem solving approach, Reema Thareja, Oxford University Press.
2. Learning Python, Mark Lutz, O'Rielly
3. Programming Python, Fourth Edition, Mark Lutz, O'Reilly Media.

References:

1. Introduction to Computation and Programming Using Python with Application to Understanding, John V. Guttag, PHI.
2. Think Python: How to think like a Computer Scientist, Allen Downey, Green Tea Press.
3. Head First Python: A Brain-Friendly Guide, Second Edition, Paul Barry, O'Reilly
4. The Python Standard Library, Python 3.6.5 documentation (Web Resource) https://docs.python.org/3/library/

	Minor Course	L	T	P	C
		3	0	2	4
BASIC WEB DESIGNING					

Course Objectives:

The objectives of this course are to acquire knowledge on the

1. How does a website work and web related terminology.
2. Web standards and W3C elements
3. Responsive Web Designing

Course Outcomes:

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

1. Learn the basic terminology related to web and web development.
2. Learn how to design static web pages by using HTML.
3. Should be able to create web pages with enhanced look and feel by Using CSS.
4. Learn to use Java Script for design thick clients and to design interactive responsive form design and validations.
5. Learn to design and host and publish websites in various domains.

UNIT - I:

Introduction to Web and Web Design Principles: Brief History of Internet, What is World Wide Web, Why create a web site, Web Standards, Web pages, Website, Web browsers and Web servers and Web protocols.

Basic principles involved in developing a web site, Planning process, Five Golden rules of web designing ,Designing navigation bar , Page design ,Home Page Layout ,Design concept.

UNIT - II:

Introduction to HTML: What is HTML, HTML Documents, Basic structure of an HTML document, Creating an HTML document, Mark up Tags, Heading-Paragraphs, Line Breaks, HTML Tags. Introduction to elements of HTML, Working with Text, Working with Lists, Tables and Frames, Working with Hyperlinks, Images and Multimedia ,Working with Forms and controls.

UNIT - III:

Introduction to Cascading Style Sheets: Concept of CSS, Creating Style Sheet, CSS Properties, CSS Styling(Background, Text Format, Controlling Fonts), Working with block elements and objects, working with Lists and Tables, CSS Id and Class ,Box Model(Introduction, Border properties, Padding Properties, Margin properties) ,CSS Advanced(Grouping, Dimension, Display, Positioning, Floating, Align, Pseudo class, Navigation Bar, Image Sprites, Attribute sector) , CSS Color ,Creating page Layout and Site Designs.

UNIT - IV:

Introduction to Java Script: What is Java Script? Basics of Java Script: Variables, functions, and Operators, select HTML elements with Java Script, Java Script Events and Event Handlers, Regular expressions and pattern matching in Java Script. Form validation using Java Script.

UNIT - V:

Introduction to Web Publishing or Hosting: Creating the Web Site, Saving the site, working on the web site, Creating web site structure, Creating Titles for web pages, Themes-Publishing web sites. Case study: Web publishing and hosting using Heroku cloud platform (https://www.heroku.com/).

Text Books		
Name of Authors	Title of the Book	Publisher
Kogent Learning Solutions Inc.	HTML 5 in simple steps	Dreamtech Press
	A beginner's guide to HTML	NCSA,14th May,2003
Murray,Tom/Lynchburg	Creating a Web Page and Web Site	College,2002
Reference Books		
	Web Designing & Architecture-Educational Technology Centre	University of Buffalo
Steven M. Schafer	HTML, XHTML, and CSS Bible, 5ed	Wiley India
John Duckett	Beginning HTML, XHTML, CSS, and JavaScript	Wiley India
Ian Pouncey, Richard York	Beginning CSS: Cascading Style Sheets for Web Design	Wiley India
Kogent Learning	Web Technologies: HTML, Javascript	Wiley India

	Minor Course	L	T	P	C
		3	0	2	4
ADVANCED WEB TECHNOLOGIES					

Course Objectives:

The objectives of this course is to acquire knowledge on the

1. This course is designed to introduce students with basic web programming experience to the advanced web programming languages and techniques associated with the World Wide Web.
2. The course will introduce web-based media-rich programming tools for creating interactive web pages.
3. The course will introduce Web Frame works like React JS and Angular JS for quick and efficient design and implementation of web applications.

Course Outcomes:

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

1. Analyze a web page and Create web pages using HTML5 and Cascading Styles sheets 3 and Boot strap.
2. Build dynamic web pages using Java Script and Write simple client-side scripts using AJAX.
3. Learn to use XML for data exchange and transfer over web and XML parsing and validation techniques.
4. Build web applications using PHP.
5. Describe a java web services.

UNIT – I

Introduction to HTML5, CSS3 and Boot strap: Basic Syntax, Standard HTML Document Structure, HTML5 tags, Audio, video, 2D canvas Drawing and animations using HTML5.
CSS 3: What is SCSS, Difference between CSS and SCSS, Introduction to SASS tool and CSS template design using Bootstrap

UNIT - II:

Java Script and DHTML: DHTML: Java Script DOM, Interactive and responsive web page designing, Positioning Moving and Changing Elements.
Java Script Web Frame works: React JS, Angular JS and Vue JS, Single Page Application (SPA) Design and Development using Angular JS.

UNIT - III:

XML: Introduction to XML, XML vs HTML, Document type Definition, XML schemas, Document object model, XSLT, DOM and SAX Approaches. AJAX A New Approach: Introduction to AJAX. Request and Response mechanism of AJAX.

UNIT - IV:

PHP Programming: Introducing PHP: Creating PHP script, Running PHP script. Working with variables and constants: Using variables, Using constants, Datatypes, Operators. Controlling program flow: Conditional statements, Control statements, Arrays, functions. Working with forms and Databases such as MySQL.

UNIT - V:

Web Services: JAX-RPC-Concepts-Writing a Java Web Service-Writing a Java Web Service Client- Describing Web Services: WSDL- Representing Data Types: XML Schema Communicating Object Data: SOAP Related Technologies-Software Installation-Storing Java Objects as Files-Databases and Java Servlets.

Text Books:

1. Programming the World Wide Web, Robet W Sebesta, 7ed, Pearson.
2. Introducing HTML5 (Voices That Matter) 2nd Edition by Bruce Lawson / Remy Sharp Lawson / Sharp,Kindle publishers.
3. Web Technologies, Uttam K Roy, Oxford
4. HTML, CSS, and JavaScript All in One: Covering HTML5, CSS3, and ES6, Sams Teach Yourself 3rd Edition, by Julie Meloni and, Jennifer Kyrnin.Pearson
5. JavaScript Frameworks for Modern Web Development: The Essential Frameworks, Libraries, and Tools to Learn Right Now 2nd ed. Edition by Sufyan bin Uzayr , Nicholas Cloud , Tim Ambler.Apress.
6. Java Web Services: Up and Running: A Quick, Practical, and Thorough Introduction 2nd Edition, Kindle Edition by Martin Kalin.

References:

1. Ruby on Rails Up and Running, Lightning fast Web development, Bruce Tate, Curt Hibbs, Oreilly(2006
2. Programming Perl, 4ed, Tom Christiansen, Jonathan Orwant, Oreilly (2012)
3. Web Technologies, HTML< JavaScript, PHP, Java, JSP, XML and AJAX, Black book, Dream Tech.
4. An Introduction to Web Design, Programming, Paul S Wang, Sanda S Katila, Cengage Learning.

	Minor Course	L	T	P	C
		3	1	0	4
COMPUTER ORGANIZATION AND ARCHITECTURE					

Course Objectives:

1. To understand the structure, function and characteristics of computer system.
2. To understand the design of the various functional units and components of computers.
3. To explain the function of each element of a memory hierarchy.

Course Outcomes:

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

1. Understand the architecture of modern computer.
2. Analyze the Performance of a computer using performance equation.
3. Understand different instruction types.
4. Calculate the effective address of an operand by addressing modes.
5. Understand how computer stores positive and negative numbers.
6. Understand how computer performs arithmetic operation of positive and negative numbers.

UNIT -I:

Basic Structure of Computers: Functional unit, Basic Operational Concepts, Bus Structures, System Software, Performance, The History of Computer Development. Data Representation: Data Types, Complements, Fixed Point Representation, Floating Point Representation.

Basic Computer Organization and Design: Instruction Codes, Computer Registers, Computer Instructions, Timing and Control, Instruction Cycle, Memory – Reference Instructions, Interrupt, Design of Basic Computer, Design of Accumulator Logic.

UNIT -II:

Machine Instruction and Programs: Instruction and Instruction Sequencing: Register Transfer Notation, Assembly Language Notation, Addressing Modes, Basic Input/output Operations, Importance of Stacks and Queues in Computer Programming Equation. Component of Instructions: Logic Instructions, Shift and Rotate Instructions, Branch Instructions.

Computer Arithmetic: Addition, Subtraction, Multiplication and Division Algorithms. Floating point Arithmetic Operations and Decimal Arithmetic Operations.

UNIT -III:

The Memory System: Memory System Consideration RAM and ROM, Flash Memory, Memory Hierarchy, Main Memory, Auxiliary Memory, Associative Memory, Cache Memory and Virtual Memory, Secondary Storage: Magnetic Hard Disks, Optical Disks.

Pipeline Processing: Parallel Processing, Pipelining, Arithmetic Pipeline, Instruction Pipeline, RISC Pipeline.

UNIT -IV:

Processing Unit: Fundamental Concepts: Register Transfers, Performing an Arithmetic or Logic Operation, Fetching a Word from Memory, Execution of Complete Instruction, Hardwired Control.

Micro Programmed Control: Microinstructions, Micro Program Sequencing, Wide Branch Addressing and Microinstructions with Next – Address Field.

UNIT -V:

Input / Output Organization:

Accessing I/O Devices, Interrupts: Interrupt Hardware, Enabling and Disabling Interrupts, Handling Multiple Devices, Direct Memory Access, Buses: Synchronous Bus, Asynchronous Bus, Interface Circuits, Standard I/O Interface: Peripheral Component Interconnect (PCI) Bus, Universal Serial Bus (USB).

Parallelism:

Instruction-Level-Parallelism – Parallel Processing Challenges – Flynn's Classification: SISD, MIMD, SIMD, SPMD and Vector Architectures, – Hardware Multithreading – Multi-Core Processors and Other Shared Memory Multiprocessors.

Text Books:

1. Computer Organization, Carl Hamacher, ZvonksVranesic, SafeaZaky, 6th Edition, McGraw Hill.
2. Computer Architecture and Organization, John P. Hayes, 3rd Edition, McGraw Hill.

References:

1. Computer Organization and Architecture – William Stallings tenth Edition, Pearson/PHI.
2. Computer System Architecture, M. Morris Mano, 3 rd Edition Pearson Education.
3. Structured Computer Organization – Andrew S.Tanenbaum, 4th Edition PHI/Pearson.
4. Fundamentals of Computer Organization and Design, Sivarama Dandamudi Springer Int.Edition.

	Minor Course	L	T	P	C
		3	1	0	4
DISTRIBUTED SYSTEMS					

Course Objectives:

1. To understand the foundations of distributed systems.
2. To learn issues related to clock Synchronization, the need for global state and remote invocation in distributed systems.
3. To learn distributed mutual exclusion and deadlock detection algorithms.

Course Outcomes:

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

1. Understand the foundations and issues of distributed systems.
2. Illustrate the various synchronization issues, global state and remote invocation for distributed systems.
3. Develop the Mutual Exclusion and Deadlock detection algorithms in distributed systems.
4. Apply the features of peer-to-peer, distributed shared memory systems and security.
5. Analyze the distributed transactions, agreement protocols and fault tolerance mechanisms in distributed systems.

UNIT- I:

Characterization of Distributed Systems: Introduction, Examples of Distributed Systems, Resource Sharing and the Web, Challenges, Relation to Computer system Components, Motivation, Relation to Parallel Systems, Message-Passing systems versus Shared Memory systems, Primitives for Distributed Communication, Synchronous versus Asynchronous executions, Design issues and Challenges.

UNIT –II:

Message Ordering and Group Communication: Message ordering paradigms, Asynchronous execution with synchronous communication, Synchronous program order on an asynchronous system, Group communication, Causal order (CO), Total order.

Global state and Snapshot Recording Algorithms: Introduction, System model and definitions, Snapshot algorithms for FIFO channels. Remote Invocation: Introduction, Design Issues for RMI, Implementation of RMI, Distributed Garbage Collection, Remote Procedure Call, Events and Notifications, Case Study: JAVA RMI.

UNIT- III:

Distributed Mutual Exclusion Algorithms: Introduction, Preliminaries, Lamport's algorithm, Ricart-Agrawala algorithm, Maekawa's algorithm, Suzuki–Kasami's broadcast algorithm.

Deadlock Detection in Distributed Systems: Introduction, System model, Preliminaries, Models of deadlocks, Knapp's Classification, Algorithms for the Single Resource Model, the AND model and the OR model.

UNIT -IV:

Peer-to-Peer Computing and Overlay Graphs: Introduction, Data indexing and overlays, Chord distributed hash table, Content addressable networks, Tapestry.

Distributed Shared Memory: Abstraction and advantages, Memory consistency models, Shared Memory Mutual Exclusion.

Security: Introduction, Overview of Security Techniques, Cryptographic Algorithms, Digital Signatures, Cryptography Pragmatics.

UNIT –V:

Distributed Transactions: Introduction, Flat and Nested Distributed Transactions, Atomic commit protocols, Concurrency control in distributed transactions. Check Pointing and **Rollback Recovery:** Introduction, Background and definitions, Issues in Failure recovery, Checkpoint-based recovery, Log-based rollback recovery, coordinated check pointing algorithm, Algorithms for asynchronous and synchronous check pointing and recovery. **Consensus and Agreement Algorithms:** Problem definition, Overview of results, Agreement in a Failure-Free system (synchronous or asynchronous).

Text Books:

1. Distributed computing: Principles, algorithms, and systems, Ajay D Kshemkalyani and Mukesh Singhal, Cambridge University Press, 2011.
2. Distributed Systems Concepts and Design, George Coulouris, Jean Dollimore and TimKindberg, 5th Edition, Pearson Education, 2012.

References:

1. Distributed Operating Systems: Concepts and Design, Pradeep K Sinha, Prentice Hall ofIndia, 2007.
2. Advanced concepts in operating systems. Mukesh Singhal and Niranjan G. Shivaratri,McGraw-Hill, 1994.
3. Distributed Systems: Principles and Paradigms, Tanenbaum A.S., Van Steen M.,Pearson Education, 2007.

E-Resources:

1. https://nptel.ac.in/courses/106/106/106106168/

	Minor Course	L	T	P	C
		3	1	0	4

CLOUD COMPUTING

Course Objective:

1. Explain the evolution of computing paradigms,
2. Differentiate cloud computing as both a service and a platform
3. Identify and evaluate different types of applications suitable for cloud deployment,

Course Outcomes:

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

1. Understand and analyze different computing paradigms
2. Understand the basics of cloud computing and different cloud deployment models.
3. Understand different cloud implementation and management strategies.
4. Understand and evaluate different cloud service models.
5. Identify, analyze and use different cloud services/applications/tools available from key cloud providers.

UNIT-I:

Computing Paradigms: High-Performance Computing, Parallel Computing, Distributed Computing, Cluster Computing, Grid Computing, Cloud Computing, Bio computing, Mobile Computing, Quantum Computing, Optical Computing, Nano computing.

UNIT-II:

Cloud Computing Fundamentals: Motivation for Cloud Computing, The Need for Cloud Computing, Defining Cloud Computing, Definition of Cloud Computing, Cloud Computing is a Service, Cloud Computing is a Platform, Principles of Cloud computing, Five Essential Characteristics, Four Cloud Deployment Models

UNIT-III:

Cloud Computing Architecture and Management: Cloud architecture, Layer, Anatomy of the Cloud, Network Connectivity in Cloud Computing, Applications on the Cloud, Managing the Cloud, Managing the Cloud Infrastructure, Managing the Cloud Application, Migrating Application to Cloud, Phases of Cloud Migration Approaches for Cloud Migration.

UNIT-IV:

Cloud Service Models: Infrastructure as a Service, Characteristics of IaaS. Suitability of IaaS, Pros and Cons of IaaS, Summary of IaaS Providers, Platformas a Service, Characteristics of PaaS, Suitability of PaaS, Pros and Cons of PaaS, Summary of PaaS Providers, Software as a Service, Characteristics of SaaS, Suitability of SaaS, Pros and Cons of SaaS, Summary of SaaS Providers, Other Cloud Service Models.

UNIT-V:

Cloud Providers and Applications: EMC, EMC IT, Captiva Cloud Toolkit, Google Cloud Platform, Cloud Storage, Google Cloud Connect, Google Cloud Print, Google App Engine, Amazon Web Services, Amazon Elastic Compute Cloud, Amazon Simple Storage Service, Amazon Simple Queue service, Microsoft, Windows Azure, Microsoft Assessment and Planning Toolkit, SharePoint, IBM, Cloud Models, IBM Smart Cloud, SAP Labs, SAP HANA

Cloud Platform, Virtualization Services Provided by SAP, Sales force, Sales Cloud, Service Cloud: Knowledge as a Service, Rackspace, VMware, Manjra soft, Aneka Platform.

Text Book:

1. Essentials of Cloud Computing, K. Chandrasekhran, CRC press.

References:

1. Cloud Computing: Principles and Paradigms, Rajkumar Buyya, James Broberg and Andrzej M. Goscinski, Wiley.
2. Distributed and Cloud Computing, Kai Hwang, Geoffery C. Fox, Jack J. Dongarra, Elsevier.
3. Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance, Tim Mather, SubraKumaraswamy, ShahedLatif, O'Reilly.

	Minor Course	L	T	P	C
		3	1	0	4

QUANTUM COMPUTING

Course Objectives

1. To introduce the fundamentals of quantum computing
2. The problem-solving approach using finite dimensional mathematics
3. Explain the principle of superposition and the concept of entanglement, highlighting their significance in quantum mechanics and computation.

Course Outcome

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

1. Understand the Basics of complex vector spaces
2. Quantum mechanics as applied in Quantum computing
3. Apply Quantum Architecture and algorithms to solve real time problems.
4. Understand and explore the models of Quantum Computer and Quantum Simulation tools
5. Analyze and implement basic quantum algorithms involving superposition, entanglement, and measurement operations.

UNIT-I:

Introduction: Complex numbers and its geometrical representations, Complex vector spaces, inner products and Hilbert spaces, Hermitian and unitary matrices, Tensor products of vector spaces Deterministic Systems

UNIT-II:

Dirac formalism, superposition of states, entanglement Bits and Qubits. Qubit operations, Hadamard Gate, CNOT Gate, Phase Gate, Z-Y decomposition, Quantum Circuit Composition, Basic Quantum circuits.

UNIT-III:

Quantum Algorithm- I: Quantum parallelism, Quantum Evolution, Deutsch's Algorithm, Deutsch-Jozsa Algorithm, Simon's periodicity algorithm.

UNIT-IV:

Quantum Algorithm- II: Grover's search algorithm, Shor's Factoring algorithm. Application of entanglement, teleportation, superdence coding.

UNIT-V

Quantum Software Development and Programming:

Quantum programming languages, Probabilistic and Quantum computations, introduction to quantum cryptography and quantum information theory.

Text Books

i. Quantum computing explained, David McMahon, Wiley-interscience, John Wiley &

Sons, 2008

ii. Quantum computing for computer scientists, Noson S. Yanofsky, Mirco A. Mannucci, Cambridge University Press 2008

Reference Books

i. Quantum computation and quantum information, Michael A. Nielsen and Isaac L. Chuang, Cambridge University Press 2010

ii. Introduction to Quantum Mechanics, 2nd Edition, David J. Griffiths, Prentice Hall New Jersey 1995