COURSE STRUCTURE

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

COURSE STRUCTURE AND SYLLABUS

For UG –R22

B. TECH - COMPUTER SCIENCE & ENGINEERING

(Applicable for batches admitted from 2022-2023)

ST.ANN'S COLLEGE OF ENGINEERING & TECHNOLOGY
(AUTONOMOUS)

COURSE STRUCTURE FOR COMPUTER SCIENCE AND ENGINEERING

Year: I Semester: I

Sl. No.	Category	Course Code	Course Title	Hours per week			CIE Marks	SEE Marks	Total Marks	Credits
				L	T	P				C
1	Basic Science course		Mathematics – I	3	0	0	30	70	100	3
2	Basic Science course		Applied Physics	3	0	0	30	70	100	3
3	Humanities and Social science		Communicative English	3	0	0	30	70	100	3
4	Engineering Science Courses		IT Workshop	1	0	4	15	35	50	3
5	Engineering Science Courses		Programming for Problem Solving using C	3	0	0	30	70	100	3
6	Humanities and Social science LAB		English Communication Skills Laboratory	0	0	3	15	35	50	1.5
7	Basic Science course (LAB)		Applied Physics Laboratory	0	0	3	15	35	50	1.5
8	Engineering Science Courses (LAB)		Programming for Problem Solving Using C Laboratory	0	0	3	15	35	50	1.5
9	Mandatory course (AICTE suggested)		Constitution of India	1	0	0	-	-	-	-
Total Credits										**19.5**

ST.ANN'S COLLEGE OF ENGINEERING & TECHNOLOGY: CHIRALA
(AUTONOMOUS)
CSE - UG – R22

Year: I Semester: II

Sl. No.	Category	Course Code	Course Title	Hours per week			CIE Marks	SEE Marks	Total Marks	Credits
				L	T	P				C
1	Basic Science course		Applied Chemistry	3	0	0	30	70	100	3
2	Basic Science course		Mathematics - II	3	0	0	30	70	100	3
3	Engineering Science Courses		Data Structures	3	0	0	30	70	100	3
4	Engineering Science Courses		Python Programming	3	0	0	30	70	100	3
5	Engineering Science Courses		Introduction to Digital Logic	3	0	0	30	70	100	3
6	Engineering Science Courses (LAB)		Data Structures using C Laboratory	0	0	3	15	35	50	1.5
7	Basic Science course (LAB)		Applied Chemistry Laboratory	0	0	3	15	35	50	1.5
8	Engineering Science Courses (LAB)		Python Programming Laboratory	0	0	3	15	35	50	1.5
9	Mandatory course (AICTE suggested)		Environmental Science	1	0	0	-	-	-	-
Total Credits										**19.5**

Year: II Semester: III

Sl. No.	Category	Course Code	Course Title	Hours per week			CIE Marks	SEE Marks	Total Marks	Credits
				L	T	P				C
1	Basic Science course	22UMT03	Mathematics III	3	0	0	30	70	100	3
2	Professional Core Course	22UCS08	Object Oriented Programming through C++	3	0	0	30	70	100	3
3	Professional Core Courses	22UCS09	Computer Organization	3	0	0	30	70	100	3
4	Professional Core Courses	22UCS10	Software Engineering	3	0	0	30	70	100	3
5	Professional Core Courses	22UCS11	Mathematical Foundations of Computer Science	3	0	0	30	70	100	3
6	Professional Core courses (LAB)	22UCS12	Object Oriented Programming through C++ Lab	0	0	3	15	35	50	1.5
7	Professional Core courses (LAB)	22UCS13	R programming Lab	0	1	2	15	35	50	2
8	Professional Core courses (LAB)	22UCS14	Software Engineering Lab	0	0	2	15	35	50	1
9	Skill oriented course*	22UCS15	Skill Oriented Course - Applications of Python-NUMPY & PANDAS	1	0	2	0	50	50	2
10	Mandatory course (AICTE suggested)	22UEN04	Essence of Indian Traditional Knowledge	2	0	0	-	-	-	-
11	Community Service project	**Community Service Project (Mandatory) (to be evaluated during IV semester)**								
Total Credits										**21.5**

Year: II Semester: IV

Sl. No.	Category	Course Code	Course Title	Hours per week			CIE Marks	SEE Marks	Total Marks	Credits
				L	T	P				C
1	Engineering Science Courses	22UCS16	Probability and Statistics	3	0	0	30	70	100	3
2	Basic Science Course /Prof core course	22UCS17	Operating Systems	3	0	0	30	70	100	3
3	Professional Core courses	22UCS18	Database management systems	3	0	0	30	70	100	3
4	Professional Core courses	22UCS19	Java Programming	3	0	0	30	70	100	3
5	Professional Core courses	22UCS20	Formal Language and Automata theory	2	1	0	30	70	100	3
6	Engineering Science Courses/Prof Core (Interdisciplinary) (LAB)	22UCS21	Java Programming Lab	0	0	3	15	35	50	1.5
7	Professional Core courses (LAB)	22UCS22	Operating Systems & Linux Lab	0	0	3	15	35	50	1.5
8	Professional Core courses (LAB)	22UCS23	DBMS Lab	0	0	3	15	35	50	1.5
9	Skill oriented course*	22UCS24	web application development using HTML, CSS, XML	1	0	2	-	50	50	2
10	Community Service project		**Community Service Project (Mandatory) (to be evaluated during IV semester)**	0	0	0	-	100	100	4
11	**Internship-I with mini project 2 Months (Mandatory) during summer vacation**									
Total Credits										**25.5**
12	Minor		Any one subject from Part-A$^{\$}$	3	0	2	30	70	100	3+1
13	Honors		Any course from the Pool, as per the opted track	4	0	0	30	70	100	4

$- Integrated Course

Year: III Semester: V

Sl. No.	Category	Course Code	Course Title	Hours per week			CIE Marks	SEE Marks	Total Marks	Credits
				L	T	P				C
1	Professional Core courses		Computer Networks	3	0	0	30	70	100	3
2	Humanities and Social Sciences		Managerial Economics and Financial Accountancy	3	0	0	30	70	100	3
3	Professional Core courses		Data Warehousing and Data Mining	3	0	0	30	70	100	3
4	Open Elective Course/Job oriented		**Open Elective-I** Open Electives offered by other departments/ (Job oriented course) Full Stack Technologies	3	0	0	30	70	100	3
5	Professional Elective courses		**Professional Elective-I** 1. Artificial Intelligence 2. Software Testing Methodologies 3. Compiler Design 4. Advanced Unix Programming	3	0	0	30	70	100	3
6	Professional Core courses Lab		Data Warehousing and Data Mining Lab	0	0	3	30	70	100	1.5
7	Professional Core courses Lab		Full Stack Technologies Lab	0	0	3	30	70	100	1.5
8	**Skill advanced course/ soft skill course***		**Skill Oriented Course** Animation course: Animation Design	0	1	2	15	35	50	2
9	Mandatory course (AICTE suggested)		Employability Skills-I	2	0	0	-	-	-	0
10	**Summer Internship-I with mini project 2 Months (Mandatory) after second year (to be evaluated during V semester)**			0	0	0	-	50	50	1.5
Total Credits										**21.5**
11	Minor		Any one subject from Part-A$	3	0	2	30	70	100	3+1
12	Honors		Any course from the Pool, as per the opted track	4	0	0	30	70	100	4

$- Integrated Course

ST.ANN'S COLLEGE OF ENGINEERING & TECHNOLOGY: CHIRALA
(AUTONOMOUS)
CSE - UG – R22

Year: III Semester: VI

Sl. No.	Category	Course Code	Course Title	Hours per week			CIE Marks	SEE Marks	Total Marks	Credits
				L	T	P				C
1	Professional Core courses		Machine Learning	3	0	0	30	70	100	3
2	Professional Core courses		Advanced data structures and algorithms	3	1	0	30	70	100	4
3	Professional Core courses		Cryptography and Network Security	3	0	0	30	70	100	3
4	Professional Elective courses		**Professional Elective-II** 1.Mobile Computing 2.Big Data Analytics 3.Object Oriented Analysis and Design 4.Distributed Systems 5. MOOCS	3	0	0	30	70	100	3
5	Open Elective Course/Job oriented		**Open Elective-II** Open Electives offered by other departments/ MEAN Stack Development *(Job Oriented Course)*	3	0	0	30	70	100	3
6	Professional Core courses Lab		Machine Learning using Python Lab	0	0	2	15	35	50	1
7	Professional Core courses Lab		MEAN Stack Development Lab	0	0	3	15	35	50	1.5
8	Professional Core courses Lab		ADSA Lab	0	0	2	15	35	50	1
9	**Skill advanced course/ soft skill course***		**Skill Oriented Course - IV** Android Lab	1	0	2	-	50	50	2
10	Mandatory course (AICTE suggested)		Employability skills-II	2	0	0	-	-	-	0
11			**Industrial/Research Internship-II (Mandatory) 2 Months during summer vacation**							
			Total Credits							**21.5**
12	**Minor**		Any one subject from Part-A[$]	**3**	**0**	**2**	**70**	**30**	**100**	**3+1**
13	**Honors**		Any course from the	**4**	**0**	**0**	**70**	**30**	**100**	**4**

			Pool, as per the opted track							
Minor course through SWAYAM				**-**	**-**	**-**	**-**	**-**	**-**	**2**

$- Integrated Course

Year: IV Semester: VII

Sl. No.	Category	Course Code	Course Title	Hours per week			CIE Marks	SEE Marks	Total Marks	Credits
				L	T	P				C
1	Professional Elective courses		1. CLOUD computing 2. Cyber security & forensics 3. social networks and semantic net 4.Quantam Computing	3	0	0	70	30	100	3
2	Professional Elective courses		1.Deep learning 2. Ethical Hacking 3. Software Testing Methodologies 4. MOOCS-NPTEL/SWAYAM	3	0	0	70	30	100	3
3	Professional Elective courses		1. Block chain Technologies 2. internet of things 3. Computer vision 4. MOOCS-NPTEL/SWAYAM	3	0	0	70	30	100	3
4	Open Elective Courses-III/ Job oriented		**Open Elective-III** Open Electives offered by other departments	3	0	0	70	30	100	3
			(Job oriented Course**)** Data Science	2	0	2	70	30	100	
5	Open Elective Courses-IV/ Job oriented		**Open Elective-IV** Open Electives offered by other departments	3	0	0	70	30	100	3
			(Job oriented Course) DevOps	2	0	2	70	30	100	
6	Humanities and Social Science Elective		1. Managerial organization behavior 2. Entrepreneurship	3	0	0	70	30	100	3
7	Skill advanced course/ soft skill course*		AWS / AZURE	1	0	2	-	50	50	2
8	**Industrial/Research Internship-II 2 Months (Mandatory) after third year (to be evaluated during VII semester**			0	0	0	-	50	50	3
Total Credits										**23**
9	**Minor**		Any one subject from Part-B$^{\$}$	**3**	**0**	**2**	**70**	**30**	**100**	**3+1**
10	**Honors**		Any course from the Pool, as per the opted track	**4**	**0**	**0**	**70**	**30**	**100**	**4**
	Minor course through SWAYAM			**-**	**-**	**-**				**2**

$- Integrated Course

Year: IV Semester: VIII

Sl. No.	Category	Course Code	Course Title	Hours per week			CIE Marks	SEE Marks	Total Marks	Credits
				L	T	P				C
1	Major Project		Major Project Work, Seminar Internship	0	0	0	60	140	200	8
Total Credits										**8**

Total credits of the course :160

Note:

1. *For integrated courses*: Theory and laboratory exams will be conducted separately, and the student concern will get credits if successfully completes both theory and laboratory. Only external exam will be conducted for Laboratory component. Credit based weightage shall be considered while awarding the grade
2. *For MOOC courses*: Based on the student's interest, student can register and complete a 12-week course one year in advance, by prior information to the concern.

Year & Sem	I Year – I Semester					
Course Code	22UMT01	L	T	P	SS	C
Course Name	MATHEMATICS-I	3	1	0	0	3

Course Objectives:

- To instruct the concept of Matrices in solving linear algebraic equations.
- To assist the students to learn the concepts of partial differentiation.
- To enlighten the learners in the concept of differential equations.
- To equip the students with standard concepts and tools at an intermediate to advanced level mathematics to develop the confidence and ability among the students to handle various real-world problems and their applications.
- To make clear the students in the concepts of Multiple Integrals.

Course Outcomes:

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

CO1: Apply matrix techniques to model and solve system of linear equations.
CO2: To apply the mean value theorems to real life problems.
CO3: Solve the differential equations related to various engineering fields.
CO4: Apply double integration techniques in evaluating areas bounded by region.
CO5: Student will learn important tools of calculus in higher dimensions. Students will become familiar with 2-dimensinal and 3-dimensional coordinate systems.

UNIT-I: System of Linear Equations, Eigen Values, Eigen Vectors (12 Hours)

Rank of a matrix by echelon form and normal form – Solving system of homogeneous and non-homogeneous linear equations – Gauss Elimination method – Eigen values and Eigen vectors and properties. Cayley-Hamilton theorem (without proof) – Reduction of a matrix to Diagonal form. Applications –Finding the inverse and power of a matrix by Cayley Hamilton theorem. Learning Resources: Text Book-1

UNIT-II: Differential Calculus (12 Hours)

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

theorem. Partial Differentiation: Introduction–Homogeneous function–Euler's theorem–Total derivative – Chain rule – Jacobian – Functional dependence –Taylor's and Mac Laurin's series expansion of functions of two variables. Applications: Maxima and Minima of functions of two variables without constraints and Lagrange's method.
Learning Resources: Text Book-1

UNIT-III: Differential Equations of First Order and First Degree (12 Hours)

Linear differential equations – Bernoulli's equations – Exact equations and equations reducible to exact form. Applications: Newton's Law of cooling – Law of natural growth and decay – Orthogonal trajectories.
Learning Resources: Text Book-1

UNIT-IV: Differential Equations of Second and Higher Order (14 Hours)

Homogeneous and Non-homogeneous differential equations of higher order with constant coefficients – with non-homogeneous term of the type e^{ax}, sin ax, cos ax, polynomials in x^n, $e^{ax}V(x)$ and $x^nV(x)$ – Method of Variation of parameters, Cauchy and Legendre's linear equations.
Applications: LCR circuits.
Learning Resources: Text Book-1

UNIT-V: Multiple Integrals (14 Hours)

Double and Triple integrals – Change of order of integration in double integrals – Change of variables to polar, cylindrical and spherical coordinates. Applications: Finding Areas and Volumes.
Learning Resources: Text Book-1

Text Books:

1. **B.S.Grewal,** Higher Engineering Mathematics, 44th Edition, Khanna Publishers.

Reference Books:

1. **Dr.T.K.V.Iyengar, Dr. B.Krishna Gandhi, S.Ranganadham, Dr. M.V.S.S.N. Prasad**, a text book of Engineering Mathematics, S.Chand Publications.
2. **N.P.Bali, Manish Goyal**, A text book of Engineering Mathematics, Lakshmi Publications
3. **B.V.Ramana,** Higher Engineering Mathematics, 2007 Edition, Tata Mc.Graw Hill Education.
4. **Erwin Kreyszig,** Advanced Engineering Mathematics, 10th Edition, Wiley-India.

Contribution of Course Outcomes (COs) towards the achievement of programme outcomes (POs) (Strong – 3, Moderate – 2, Weak – 1)

CO – PO MAPPING:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	3	2	-	-	-	-	-	-	-	3
CO2	2	2	2	2	-	-	-	-	-	-	-	3
CO3	2	2	2	2	-	-	-	-	-	-	-	3
CO4	3	3	2	2	-	-	-	-	-	-	-	3
CO5	3	3	3	2	-	-	-	-	-	-	-	3

Year & Sem	I Year – I Semester					
Course Code	22UPH01	L	T	P	SS	C
Course Name	APPLIED PHYSICS	3	1	0	0	3

This course centers on unifying essential theoretical concepts of Physics governing the physical properties of materials to interpret them from the perspective of engineering and technical applications.

COURSE DESCRIPTION AND OBJECTIVES:

This course provides seamless consolidation of basic principles of Physics and applications. It emphasizes on modern technological advancement relevant to the latest developments in the fields of science, engineering, and technology and to have an insight into Dielectric and magnetic materials, principles of quantum mechanics, and electron dynamics of solids from the perspective of optoelectronic devices.

1. To identify the importance of the optical phenomenon i.e. interference, diffraction and polarization related to its Engineering applications

2. Understand the mechanism of emission of light, utilization of lasers as coherent light sources for low and high energy applications, study of propagation of light through optical fibers and their implications in optical communications.

3. Enlightenment of the concepts of Quantum Mechanics and to provide fundamentals of de Broglie matter waves, quantum mechanical wave equation and its application, the importance of free electron theory for metals and band theory for crystalline solids. Metals- Semiconductors-Insulators concepts utilization of transport phenomenon of charge carriers in semiconductors.

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

5. To Understand the physics of optoelectronic devices and working mechanism of display devices.
6. **Course Outcomes:**

1. Select the concepts of Physical Optics in view of engineering applications. Apply the knowledge of dielectric and magnetic materials to analyse them.
2. Grade the wavelengths of Lasers for suitable applications in the field of industry, medicine and communication and foster the knowledge on optical fibers.
3. Appraise electron dynamics based on quantum principles.
4. Choose dielectric and magnetic material to demonstrate the functioning of electric and electronic devices.
5. Judge the performance of optoelectronic devices based on their construction.

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

Interference:

Principle of superposition –Interference of light - Interference in thin films (Reflection Geometry) & applications - Colors in thin films- Newton's Rings- Determination of wavelength and refractive index. **Diffraction:** Introduction - Fresnel and Fraunhofer diffraction - Fraunhofer diffraction due to single slit, double slit - N-slits (Qualitative) – Diffraction Grating - Dispersive power and resolving power of Grating (Qualitative). **Polarization:** Introduction-Types of polarization - Polarization by reflection, refraction and Double refraction - Nicol's Prism -Half wave and Quarter wave plates.

Unit 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)
- Analyze the differences between interference and diffraction with applications (L4)
- Illustrate the concept of polarization of light and its applications (L2)
- Classify ordinary polarized light and extraordinary polarized light (L2)

Unit-II: Lasers and Fiber optics **8hrs**

Lasers: Introduction – Characteristics of laser – Spontaneous and Stimulated emissions of radiation – Einstein's coefficients – Population inversion – Lasing action - Pumping mechanisms – Ruby laser – He-Ne laser - Applications of lasers. **Fiber optics:** Introduction –Principle of optical fiber- Acceptance Angle - Numerical Aperture - Classification of optical fibers based on refractive index profile and modes – Propagation of electromagnetic wave through optical fibers - Applications.

Unit Outcomes:

- The students will be able to Understand the basic concepts of LASER light Sources (L2)
- Apply the concepts to learn the types of lasers (L3)

- Identifies the Engineering applications of lasers (L2)
- Explain the working principle of optical fibers (L2)
- Classify optical fibers based on refractive index profile and mode of propagation (L2)
- Identify the applications of optical fibers in various fields (L2)

Unit III: Quantum Mechanics, Free Electron Theory and Band 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– Equation for electrical conductivity based on quantum free electron theory-Fermi-Dirac distribution- Density of states (3D) - Fermi energy. **Band theory of Solids:** Bloch's Theorem (Qualitative) - Kronig - Penney model (Qualitative)- E vs K diagram - v vs K diagram - effective mass of electron – Classification of crystalline solids–concept of hole.

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)
- Explain the importance of K-P model¬ Classify the materials based on band theory (L2)
- Apply the concept of effective mass of electron (L3)

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

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

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)
- Explain the applications of dielectric and magnetic materials (L2)
- Apply the concept of magnetism to magnetic data storage devices (L3)

Unit V: Semiconductors, Opto Electronic Devices and Nano Materials 10 hrs

Semiconductors: Bands in solids- Valence and conduction band, effective mass (Qualitative), Intrinsic and extrinsic semiconductors-P type and N type, Donor and acceptor levels (Qualitative), Determination of energy gap in semiconductors. Drift and Diffusion currents, Einstein relations, Direct and indirect semiconductors, **Opto electronic Devices:** Photo voltaic effect, Solar cell, Photo detectors, Photodiodes-PIN and APD, Principle and working of LED, Liquid crystal display (LCD), Applications of opto electronic devices.

Text books:

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

Reference Books:

1. Fundamentals of Physics – Halliday, Resnick and Walker, John Wiley &Sons
2. Engineering Physics by M.R.Srinivasan, New Age international publishers (2009).
3. Shatendra Sharma, Jyotsna Sharma, "Engineering Physics", Pearson Education, 2018
4. Engineering Physics - Sanjay D. Jain, D. Sahasrabudhe and Girish, University Press
5. Semiconductor physics and devices- Basic principle – Donald A, Neamen, McGraw Hill
6. B.K. Pandey and S. Chaturvedi, Engineering Physics, Cengage Learning

Contribution of Course Outcomes (COs) towards the achievement of programme outcomes (POs) (Strong – 3, Moderate – 2, Weak – 1)

CO – PO MAPPING:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	3	-	-	-	-	-	-	-	-	-
CO2	3	3	3	-	-	-	-	-	-	3	-	-
CO3	3	-	3	3	3	-	-	-	-	-	-	-
CO4	-	-	-	-	-	-	-	-	-	-	-	-

CO5	3	3	3	3	-	-	-	-	-	3	-	-

Year & Sem	I Year – I Semester					
Course Code	22UEN01	L	T	P	SS	C
Course Name	COMMUNICATIVE ENGLISH	3	1	0	0	3

Introduction

With the growing importance of English for global communication and the emphasis on training the learners to gain communicative competence, the syllabus is designed to develop linguistic and communicative competence of the engineering students. The major focus of the syllabus is to enhance the **Help students develop effective listening skills so that they can** communicative ability, with the focus on the language skills, grammar, vocabulary of the learners and to improve the learner's ability to use English language effectively in social, academic and professional contexts. There is a shift from learning about the language to using the language. Thereby enables the learner to appear confidently for international language qualification tests like IELTS, TOEFL, BEC Etc.

Course Objectives:

- understand academic lectures and native English speakers' speech.
- Encourage the development of speaking abilities by taking part in exercises like role-playing, dialogues, and organized talks / oral presentations.
- Pay special attention to effective reading techniques for understanding a range of academic literature and real-world resources.
- Introduce useful writing techniques and illustrate them by summarizing, composing essays with a clear structure, recording and reporting relevant information.
- Increase vocabulary and grammatical knowledge, and promote proper use of words both in speech and writing.

Course Outcomes:

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

- Comprehend social or transactional discussions presented by native English speakers and recognize the context, subject, and specific information.
- Introduce one self and others and engage in general conversation about well-known subjects.
- Employ suitable strategies for skimming and scanning to get the general

idea of a text and locate specific information.

- Recognize paragraph structure and be able to match beginnings/endings/headings withparagraphs
- Form sentences using proper grammatical structures and correct word forms.

Unit- 1

Lesson-1: The Scare Crow by Satyajit Ray from Panorama, a course on reading, Oxford publications. Listening: Listening to short audio texts and identifying the topic. Listening to prose and conversations. **Speaking:** Asking and answering general questions on familiar topics such as home, family, work studies and interests. Self-introduction and introducing others. **Reading:** Skimming text to get the main idea. Scanning to look for specific pieces of information. **Writing:** Paragraph writing (specific topics) using suitable cohesive devices, linkers, sign posts and transition signals; mechanics of writing - punctuation, capital letters.

Vocabulary: Synonyms and Antonyms, Affixes.

Grammar: Content words and function words, word forms.

Unit-2

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

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

Speaking: Discussion in pairs / small groups on specific topics followed by short structured talks. Functional English: Greetings and leave takings. **Reading**: Identifying sequence of ideas; recognizing verbal techniques that help to link the ideas in a paragraph together. **Writing**: Summarizing - identifying main idea and rephrasing what is read; avoiding redundancies and repetitions. **Vocabulary**: Synonyms and Antonyms, Root words **Grammar**: Parts of Speech.

Unit- 3

Lesson-1: Telephone Conversation by Wole Soyinka

Listening: Listening for global comprehension and summarizing what is listened to, both in speaking and writing. **Speaking:** Discussing specific topics in pairs or small groups and reporting what is discussed. Functional English: Complaining and Apologizing. **Reading**: Reading a text in detail by making basic inferences - recognizing and interpreting specific context clues; strategies to use text clues for comprehension. Critical reading. **Writing**: Letter writing-types, format and principles of letter writing. E-mail etiquette, Writing CV's.

Vocabulary: Synonyms and Antonyms, Word Formation

Grammar: **Verbs, Subject Verb agreement, Common Errors.**

Unit 4

Lesson-1: Water the Elixir of life by C.V.Raman

Listening: Making predictions while listening to conversations/ transactional dialogues without video (only audio); listening to audio-visual texts. **Speaking:** Role plays for practice of conversational English in academic contexts (formal and informal) - asking for and giving information/directions. Functional English: Permissions, Requesting, Inviting. **Reading**: Studying the use of graphic elements in texts to convey information, reveal trends/patterns/relationships, communicative process or display complicated data.

Writing: Information transfer; describe, compare, contrast, identify significance/trends based on information provided in figures/charts/graphs / tables, writing for media. **Vocabulary**: Synonyms and Antonyms, Phrasal verbs.

Grammar: Tenses, correction of sentences.

Unit 5

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

Listening: Identifying key terms, understanding concepts and interpreting the concepts both in speaking and writing, TEDX Videos. **Speaking:** Formal oral presentations on topics from academic contexts - without the use of PPT slides. Functional English: Suggesting/Opinion giving. **Reading**: Reading for comprehension. RAP Strategy Intensive reading and Extensive reading techniques. **Reading for Writing**: Writing academic proposals- writing research articles: format and style. **Vocabulary**: Synonyms and Antonyms, Idioms and Phrases. **Grammar**: Voices, Degrees of comparison & Reported speech.

Suggested books:

1. **Infotech English**", Maruthi Publications.
2. **"Panorama, a course on reading", Oxford publications**
3. Bailey, Stephen. Academic writing: A handbook for international students. Routledge, 2014.
4. Chase, Becky Tarver. Pathways: Listening, Speaking and Critical Thinking. Heinley ELT; 2nd Edition, 2018.
5. Skillful Level 2 Reading & Writing Student's Book Pack (B1) Macmillan Educational.
6. Hewings, Martin. Cambridge Academic English (B2). CUP, 2012

7. https://onlinecourses.nptel.ac.in/noc20_hs19/preview
8. https://nptel.ac.in/courses/109106094
9. https://news.stanford.edu (Steve Jobs' Speech)

Contribution of Course Outcomes (COs) towards the achievement of programme outcomes (POs) (Strong – 3, Moderate – 2, Weak – 1)

CO – PO MAPPING:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	-	-	-	-	-	-	-	1	1	3	1	1
CO2	-	-	-	-	-	-	-	-	-	3	-	-
CO3	-	-	-	-	-	-	-	-	-	-	-	-
CO4	-	-	-	-	-	-	-	-	-	2	-	-
CO5	-	-	-	-	-	-	-	-	-	2	-	-

Year & Sem	I Year – I Semester					
Course Code	22UCS01	L	T	P	SS	C
Course Name	PROGRAMMING FOR PROBLEM SOLVING USING C	3	1	0	0	3

Course Objectives:

The objectives of Programming for Problem Solving Using C are

- To learn about the computer systems, computing environments, developing of a computerprogram and Structure of a C Program
- To gain knowledge of the operators, selection, control statements and repetition in C
- To learn about the design concepts of arrays, strings
- To assimilate about pointers, dynamic memory allocation
- To assimilate about File, I/O and significance of functions

Course Outcomes:

Upon the completion of the course the student will learn

CO1: To write algorithms and to draw flowcharts for solving problems and to convert flowcharts/algorithms to C Programs, compile and debug programs

CO2: To use different operators, data types and write programs that use two-way/ multi-way selection

CO3: To select the best loop construct for a given problem

CO4: To design and implement programs to analyze the different pointer applications

CO5: To decompose a problem into functions and to develop modular reusable code and to apply File I/O operations

UNIT-I

Introduction to Computers: Computer Systems - Block Diagram of Computer, Hardware, Software, Algorithms, Flow Charts, Pseudocode **Introduction to the C Language:** Background, C Programs, Identifiers, Types, Variable, Constants, Input/output, Programming Examples. **Structure of a C Program:** Expressions, types of expressions, Precedence and Associativity, Evaluating Expressions, Type Conversion Statements, Simple Programs.

UNIT-II

Bitwise Operators: Logical Bitwise Operators, Shift Operators, Programming Examples. **Selection & Making Decisions:** Logical Data and Operators, Two Way Selection, Multi way Selection, Programming examples. **Repetition:** Concept of Loop, Pretest and Post-test Loops, Initialization and Updating, Event and Counter Controlled Loops, Loops in C, Looping Applications, Programming Examples.

UNIT-III

Arrays: Concepts, Using Array in C, Array Application, Two Dimensional Arrays,

Multidimensional Arrays, Programming Examples. **Strings:** String Concepts, C String, String Input / Output Functions, Arrays of Strings, String Manipulation Functions, Programming Examples.

UNIT-IV

Functions: Designing, Structured Programs, Function in C, User Defined Functions, Types of functions, Standard Functions, Passing Array to Functions and Passing Pointers to Functions, Recursion, Scope - Global Scope, Local Scope, Function Scope, and Storage Classes. **Pointers:** Introduction - Definition, Declaration, Initialization, Accessing, Benefits of Pointers, Why Pointers, Pointers to pointers, Arrays, and Pointers, Pointer Arithmetic and Arrays, Memory Allocation Function, Array of Pointers, Programming Application.

UNIT-V

Structures and Union: Structure- Definition, Declaration, Accessing, Initialization, Arrays in Structures, Array of Structures, Structure Pointers, structures and functions, Unions and Programming Examples. **Files:** Files, Streams, Types of Files- Text and Binary Files, Standard Library Input / Output Functions, Formatting Input / Output Functions, Character Input / Output Functions, Programming Examples

Text Books:

1) Programming for Problem Solving, Behrouz A. Forouzan, Richard F.Gilberg, CENGAGE.
2) The C Programming Language, Brian W.Kernighan, Dennis M. Ritchie, 2e, Pearson.

Reference Books:

1) Computer Fundamentals and Programming, Sumithabha Das, Mc Graw Hill.
2) Programming in C, Ashok N. Kamthane, Amit Kamthane, Pearson.
3) Computer Fundamentals and Programming in C, Pradip Dey, Manas Ghosh, OXFORD.

Contribution of Course Outcomes (COs) towards the achievement of programme outcomes (POs) (Strong – 3, Moderate – 2, Weak – 1)

CO – PO MAPPING:

Course	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3											
CO2	3	1	2									
CO3	3		1									
CO4		2	3	2	3							
CO5		1		2	3							

Year & Sem	I Year – I Semester					
Course Code	22UEN02	L	T	P	SS	C
Course Name	ENGLISH COMMUNICATION SKILLS LAB	0	0	3	0	1.5

Course Objectives

- Learners learn the basics of phonetics- recognize phonetic symbols and facilitates the learners' use of dictionary for pronunciation.
- To enhance the articulation of the sounds and pronunciation of words.
- To improve the communication skills and clarity of speech.
- To enhance effective communication skills.
- Enables learners to speak and communicate confidently.

Course Outcomes:

By the end of the semester the learners develop

1. Proper and accurate articulation of the sounds by following standard pronunciation of words and communicate intelligibly.
2. Speaking fluently with neutral accent.
3. Clarity of speech.
4. To communicate in various contexts using choice of appropriate expressions.
5. To acquire several communicative functions. Thereby enable to interact in different social and work situations.

The course material is divided into five units.

Unit 1:

Introduction and importance of phonetics. Letters and Sounds, Sounds of English (Consonant Sounds, Vowel Sounds)

Unit 2

Pronunciation and pronunciation rules, Plural and past tense marker rules.

Unit 3:

Syllable, word stress, stress in mono, di, and poly syllabic words, stress in compound words, contrastive, word stress, Rhythm and Intonation.

Unit 4:

Just A Minute (JAM)

Unit 5:

Group Discussions and Interview Skills

Suggested books:

1. Infotech English, Maruthi Publications (with Compact Disc).
2. English Pronunciation in use- Mark Hancock, Cambridge University Press.
3. English Pronunciation Dictionary- Daniel Jones, Cambridge University Press.
4. English Phonetics for Indian Students- P. Bala Subramanian, Mac Millan Publications.
5. Strengthen your communication skills by Dr M Hari Prasad, Dr Salivendra Raju
6. English for Professionals by Prof Eliah, B.S Publications, Hyderabad.
7. Cornerstone, Developing soft skills, Pearson Education Technical Communication- Meenakshi Raman, Sangeeta Sharma, Oxford University Press.
8. https://nptel.ac.in/courses/109106067

Contribution of Course Outcomes (COs) towards the achievement of programme outcomes (POs) (Strong – 3, Moderate – 2, Weak – 1)

CO – PO MAPPING:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	-	-	-	-	-	-	-	-	-	3	-	-
CO2	-	-	-	-	-	-	-	-	-	3	-	-
CO3	-	-	-	-	-	-	-	-	-	3	-	-
CO4	-	-	-	-	-	-	-	-	2	3	-	1
CO5	-	-	-	-	-	-	-	-	-	3	-	-

Year & Sem	I Year – I Semester					
Course Code	22UPH02	L	T	P	SS	C
Course Name	APPLIED PHYSICS LABORATORY	0	0	3	0	1.5

(Any 10 of the following listed experiments)

List of Applied Physics Experiments

1. Determination of thickness of thin object by wedge method.
2. Determination of radius of curvature of a given plano convex lens by Newton's rings.
3. Determination of wavelengths of different spectral lines in mercury spectrum using diffraction grating in normal incidence configuration.
4. Determination of dispersive power of the prism.
5. Determination of dielectric constant using charging and discharging method.
6. Study the variation of B versus H by magnetizing the magnetic material (B-H curve).
7. Determination of numerical aperture and acceptance angle of an optical fiber.
8. Determination of wavelength of Laser light using diffraction grating.
9. Estimation of Planck's constant using photoelectric effect.
10. Determination of the resistivity of semiconductor by four probe method.
11. To determine the energy gap of a semiconductor using p-n junction diode.
12. Magnetic field along the axis of a current carrying circular coil by Stewart &Gee's Method
13. Determination of Hall voltage and Hall coefficient of a given semiconductor using Hall Effect.
14. Measurement of resistance of a semiconductor with varying temperature.
15. Resistivity of a Superconductor using four probe method & Meissner effect

References:

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

Course learning objectives

1. To gain practical knowledge by applying the experimental methods to correlate with the Physics theory. 2. To learn the usage of electrical and optical systems for

various measurements.

Year & Sem	I Year – I Semester

3. Apply the analytical techniques and graphical analysis to the experimental data.
4. To develop intellectual communication skills and discuss the basic principles of scientific concepts in a group.

Course Outcomes (COs)

At the end of the course, the student will be able to
1. Apply the various procedures and techniques for the experiments.
2. Use the different measuring devices and meters to record the data with precision.
3. Apply the mathematical concepts/equations to obtain quantitative results.
4. Develop basic communication skills through working in groups in performing the laboratory experiments and by interpreting the results.

Contribution of Course Outcomes (COs) towards the achievement of programme outcomes (POs) (Strong – 3, Moderate – 2, Weak – 1)

CO – PO MAPPING:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	-	-	-	3	-	-	-	-	-	-	-
CO2	3	-	-	-	3	-	-	-	-	-	-	-
CO3	2	-	-	-	3	-	-	-	-	-	-	-
CO4	3	-	-	-	3	-	-	-	-	-	-	-

Course Code	22UCS02	L	T	P	SS	C
Course Name	PROGRAMMING FOR PROBLEM SOLVING USING C LAB	0	0	3	0	1.5

Course Objectives**:**

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

Course Outcomes:

By the end of the Lab, the student

CO1: Gains Knowledge on various concepts of a C language.

CO2: Able to draw flowcharts and write algorithms.

CO3: Able design and development of C problem solving skills.

CO4: Able to design and develop modular programming skills.

CO5: Able to trace and debug a program

Exercise 1:

1. Write a C program to print a block F using hash (#), where the F has a height of sixcharacters and width of five and four characters.
2. Write a C program to compute the perimeter and area of a rectangle with a height of 7inches and width of 5 inches.

Exercise 2:

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

Exercise 3:

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

Exercise 4:

1. Write a program in C to display the n terms of harmonic series and their sum.
   1 + 1/2 + 1/3 + 1/4 + 1/5 ... 1/n terms.

2. Write a C program to check whether a given number is an Armstrong number or not.

Exercise 5:

1. Write a program to display all prime numbers less than n
2. Write a program to display the following output format

```
1
2  2
3  3  3
4  4  4  4
5  5  5  5  5
```

Exercise 6:

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

Exercise 7:

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

Exercise 8:

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

Exercise 9:

1. Write a c program to read and display the details of an employee using structure
2. Write a c program to demonstrate array of structures

Exercise 10:

1. Write a program in C to demonstrate the use of & (address of) and *(value at address)operator.
2. Write a C program to find sum of n elements entered by user using pointers
3. Write a C Program to Store Information Using Structures with Dynamically MemoryAllocation

Exercise 11:

1. Write a program in C to swap elements using call by reference
2. Find factorial of given number using recursion
3. Write a program in C to get the largest element of an array using the function

Exercise 12:

1. Write a program in C to append multiple lines at the end of a text file.
2. Write a program in C to copy a file in another name.
3. Write a program in C to remove a file from the disk

CO – PO MAPPING:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	3	1	2							1
CO2	3			2	3							1
CO3		3		2	3							1
CO4	3	2		2	3							1
CO5	3		2	2	3							1

Contribution of Course Outcomes (COs) towards the achievement of programme outcomes (POs) (Strong – 3, Moderate – 2, Weak – 1)

Year & Sem	I Year – I Semester					
Course Code	22UCS03	L	T	P	SS	C
Course Name	IT WORKSHOP	1	0	4	0	3

Course Objectives:

The objective of this lab is to

- Explain the internal parts of a computer, peripherals, I/O ports, connecting cables
- Demonstrate basic DOS commands
- Describe about Antivirus tools
- Demonstrate Office Tools such as Word processors, Spreadsheets and Presentationtools

Course Outcomes:

CO1: By the end of this lab the student is able to Assemble and disassemble components of a Personal Computer

CO2: By the end of this lab the student is familiar with DOS commands

CO3: By the end of this lab the student is familiar Viruses

CO4: By the end of this lab the student is able to work on Word, Power Point and MS Excel

List of Experiments:

UNIT-1

Block diagram of a computer, Identification of peripherals of a PC, Laptop, Server, Smart phones, prepare a report containing the block diagram along with the configuration of each component and its functionality, Input/Output devices, I/O ports and Interfaces, Main Memory, Cache memory and Secondary Storage Devices, Digital Storage Basics, Networking Components and Speeds.

Experiment -1: Identification of peripherals.

Experiment -2: Assembling, Dissembling of a computer.

UNIT-2

Software: Definition, Software types, Application Software, System Software**.**

Experiment -1: DOS Commands.

UNIT-3

MSWord: Creating a Document, Formatting, Bullets and Numbering, Page Settings, Headerand Footer, Insert Word Art, Clip Art, Tables.

Experiment -1: Demonstrate and practice on word Formatting **(**Bold, Italic, Underline, Alignments, Fonts, Sizes, Headings etc.,**)**

Experiment -2: Demonstrate and Practice on Page Settings, Margins, Header and Footer**.Experiment -3:** Demonstrate and Practice on WordArt, ClipArt.

Experiment -4: Demonstrate and Practice on Table Creation.

UNIT-4

MS Excel: Create work sheet and work book, Search for data with in a Web, Insert Rows, Columns, Hiding of Rows and Columns, Renaming of Worksheet, Adjust Row Height and Column Width, Create Tables.

Experiment -1: Create a Table, Perform sum and average of a sheet.

Experiment -2: Adjust Row height and Column width of a Table as per the Requirements

UNIT-5

MS Power Point: Create and Manage Presentation, Slide transition and Animation, WWW,Web Browser, Virus, Antivirus, Creating mails.

Experiment -1: Create a PPT on a Topic of your Choice.

Experiment -2: Create a mail id with your Roll Number

Contribution of Course Outcomes (COs) towards the achievement of programme outcomes (POs) (Strong – 3, Moderate – 2, Weak – 1)

CO – PO MAPPING:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1									1
CO2	3	1	1									1
CO3	3	1	1	2	3							2
CO4	3	3	3	2	3							2

Year & Sem	I Year – I Semester					
Course Code	22UEN03	L	T	P	SS	C
Course Name	CONSTITUTION OF INDIA	2	0	0	0	0

Course Objectives:

- To Enable the student to understand the importance of Constitution
- To understand the structure of Executive, Legislature and Judiciary
- To understand philosophy of Fundamental Rights and Duties
- To understand the autonomous nature of constitutional bodies like Supreme Court and High Court Controller and Auditor General of India and Election Commission of India.
- To understand the Central and State relation Financial and Administrative.

Course Outcomes:
At the end of the semester/course, the student will be able to have a clear knowledge on the following:

- Understand Historical Background of the Constitution Making and its importance for building a Democratic India.
- Understand the functioning of three wings of the Government i.e., Executive, Legislative and Judiciary.
- Understand the value of the Fundamental Rights and Duties for becoming good citizen of India.
- Analyze the decentralization of power between Central, State and local Self-Government.
- Apply the knowledge in strengthening of the constitutional institutions like CAG, Election Commission and UPSC for sustaining Democracy.

1. Know the sources, features and principles of Indian Constitution.
2. Learn about Union Government, State Government and its Administration.
3. Get acquainted with Local Administration and Panchayati Raj.
4. Be aware of basic concepts and developments of Human Rights.
5. Gain knowledge on Roles and Functioning of Election Commission.

UNIT-I
Introduction to Indian Constitution: Constitution meaning of the term, Indian Constitution - Sources and Constitutional History, Features - Citizenship, Preamble, Fundamental Rights and Duties, Directive Principles of State Policy.

Learning outcomes:

After completion of this unit student will

- Understand the concept of Indian Constitution
- Apply the knowledge on Directive Principle of State Policy
- Analyze the History, Features of Indian Constitution
- Evaluate Preamble Fundamental Rights and Duties

UNIT-II

Union Government and its Administration Structure of the Indian Union: Federalism, Centre- State relationship, President: Role, Power and Position, PM and Council of Ministers, Cabinet and Central Secretariat, Lok Sabha, Rajya Sabha, The Supreme Court and High Court: Powers and Functions

Learning outcomes: -After completion of this unit student will

- Understand the structure of Indian Government
- Differentiate between the State and Central Government
- Explain the role of President and Prime Minister
- Know the Structure of Supreme Court and High court

UNIT-III

State Government and its Administration Governor - Role and Position - CM and Council of Ministers, State Secretariat: Organization, Structure and Functions

Learning outcomes: -After completion of this unit student will

- Understand the structure of State Government
- Analyze the role of Governor and Chief Minister
- Explain the role of State Secretariat
- Differentiate between Structure and Functions of State Secretariat

UNIT-IV

A. Local Administration - District's Administration Head - Role and Importance, Municipalities - Mayor and role of Elected Representative - CEO of Municipal Corporation Pachayati Raj: Functions PRI: Zilla Panchayat, Elected Officials and their roles, CEO Zilla Panchayat: Block level Organizational Hierarchy - (Different departments), Village level - Role of Elected and Appointed officials - Importance of grass root Democracy

Learning outcomes: -After completion of this unit student will

- Understand the Local Administration
- Compare and contrast District Administration role and importance
- Analyze the role of Mayor and elected representatives of Municipalities
- Evaluate Zilla panchayat block level organization

UNIT-V

Election Commission: Election Commission- Role of Chief Election Commissioner and Election Commissionerate State Election Commission: Functions of Commissions for the welfare of SC/ST/OBC and women

Learning outcomes: -After completion of this unit student will

- Know the role of Election Commission apply knowledge
- Contrast and compare the role of Chief Election Commissioner and Commissionerate
- Analyze role of State Election Commission
- Evaluate various commissions of viz SC/ST/OBC and women

References:

1. Durga Das Basu, Introduction to the Constitution of India, Prentice – Hall of India Pvt.Ltd.. NewDelhi
2. Subash Kashyap, Indian Constitution, National Book Trust
3. J.A. Siwach, Dynamics of Indian Government & Politics
4. D.C. Gupta, Indian Government and Politics
5. H.M.Sreevai, Constitutional Law of India, 4th edition in 3 volumes (Universal Law Publication)
6. J.C. Johari, Indian Government and Politics Hans
7. J. Raj Indian Government and Politics
8. M.V. Pylee, Indian Constitution Durga Das Basu, Human Rights in Constitutional Law, Prentice –Hall of India Pvt. Ltd. New Delhi
9. Noorani, A.G., (South Asia Human Rights Documentation Centre), Challenges to Civil Right), Challenges to Civil Rights Guarantees in India, Oxford University Press 2012

E-resources:

1. nptel.ac.in/courses/109104074/8
2. nptel.ac.in/courses/109104045/
3. nptel.ac.in/courses/101104065/
4. www.hss.iitb.ac.in/en/lecture-details
5. www.iitb.ac.in/en/event/2nd-lecture-institute-lecture-series-indian-constitution

Contribution of Course Outcomes (COs) towards the achievement of programme outcomes (POs) (Strong – 3, Moderate – 2, Weak – 1)

CO – PO MAPPING:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	-	-	-	-	-	2	-	-	-	-	-	-
CO2	-	-	-	-	-	1	-	-	-	-	-	-
CO3	-	-	-	-	-	-	-	2	-	-	-	-
CO4	-	-	-	-	-	-	-	1	-	-	-	-
CO5	-	-	-	-	-	1	-	-	-	-	-	-

Year & Sem	I Year – II Semester					
Course Code	22UMT02	L	T	P	SS	C
Course Name	MATHEMATICS-II	3	1	0	0	3

Course Objectives:

- To illuminate the different numerical methods to solve nonlinear algebraic equations.
- To give a definition of Interpolation as it relates to mapping / surveying.
- To familiarize the Laplace, transform techniques in solving the Differential Equations.
- To familiarize the Fourier Series expansions for periodic functions.
- To furnish the learners with basic concepts Fourier Transform techniques to lead them into advanced level by handling various real-world applications.

Course Outcomes:

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

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

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

CO3: Apply the Laplace transform for solving differential equations.

CO4: Find or compute the Fourier series of periodic signals.

CO5: Apply integral expressions for the forwards and inverse Fourier transform to arrange of non - Periodic wave forms.

UNIT-I Iterative Methods (12 Hours)

Introduction– Bisection method–Secant method – Method of false position– Iteration method –Newton - Raphson method (One variable only) – Jacobi and Gauss-Seidel methods for solving system of equations numerically.

Learning Resources: Text Book – 1

UNIT-II Interpolation (12 Hours)

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

Learning Resources: Text Book - 1

UNIT-III Laplace Transforms and Inverse Laplace Transforms (14 Hours)

Laplace transforms –Definition and Laplace transforms of some certain functions – Shifting theorems – Transforms of derivatives and integrals – Unit step function –Dirac's delta function Periodic function – Inverse Laplace transforms – Convolution theorem (without proof).Applications: Solving ordinary differential equations (initial value problems) using Laplace transforms.

Learning Resources: Text Book – 1

UNIT IV Fourier Series (12 Hours)

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

Learning Resources: Text Book – 1

UNIT V Fourier Transforms (14 Hours)

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

Learning Resources: Text Book - 1

Text Books:

1. **B.S.Grewal,**Higher Engineering Mathematics, 44th Edition, Khanna Publishers.

Reference Books:

1. **Dr.T.K.V. Iyengar, Dr. B. Krishna Gandhi, S. Ranganadham, Dr. M.V.S.S.N. Prasad**, A text book of Engineering Mathematics, S. Chand Publications.
2. **B.V. Ramana,** Higher Engineering Mathematics, 2007 Edition, Tata Mc. Graw Hill Education.
3. **N.P. Bali, Manish Goyal**, A text book of Engineering Mathematics, Lakshmi Publications
4. **Erwin Kreyszig,** Advanced Engineering Mathematics, 10th Edition, Wiley-India.

Contribution of Course Outcomes (COs) towards the achievement of programme outcomes (POs) (Strong – 3, Moderate – 2, Weak – 1)

CO – PO MAPPING:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	3	2	-	-	-	-	-	-	-	3
CO2	3	2	3	2	-	-	-	-	-	-	-	3
CO3	2	2	2	2	-	-	-	-	-	-	-	2
CO4	2	3	3	2	-	-	-	-	-	-	-	2
CO5	3	3	2	2	-	-	-	-	-	-	-	3

Year & Sem	I Year – II Semester					
Course Code	22UCH01	L	T	P	SS	C
Course Name	APPLIED CHEMISTRY	3	1	0	0	3

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

COURSE OBJECTIVES

1. Importance of usage of plastics in household appliances and composites (FRP) in aerospace and automotive industries.
2. Outline the basics for the construction of electrochemical cells, batteries and fuel cells. Understand the mechanism of corrosion and how it can be prevented
3. Explain the preparation of nanomaterials, engineering applications of nanomaterials, superconductors, preparation of semiconductors and applications of Hall effect.
4. Importance of Non-conventional Energy Resources, its design and working along with Fuel cells and Spectroscopic techniques.
5. Outline the basics of computational chemistry and molecular switches.

COURSE OUTCOMES:

CO1. Analyze types of plastics, methods of fabrication, the different types of composite plastic materials and interpret the mechanism of conduction in conducting polymers.

CO2. Understand the theory of construction of electrodes, batteries in redesigning new engineering products and categorize the reasons for corrosion and study methods to control corrosion.

CO3. Understand synthesis of nanomaterials for modern advances of engineering technology analyze the applications of super conductors. Summarize the preparation of semiconducting; analyse the applications of Hall effect.

CO4. Analyze different models of energy harnessing from different natural sources and also working and applications of Fuel cells and Spectroscopic Techniques.

CO5. Understand the knowledge of computational chemistry and molecular machines.

UNIT I: POLYMER TECHNOLOGY (**8hrs)**

Polymerization: -Introduction, types and methods of polymerization (emulsion and suspension), mechanical properties.

Plastics: Types, Compounding, Fabrication (compression, injection, blowing and extrusion moldings), preparation, properties and applications (PVC, polycarbonates and Bakelite), mention some examples of plastic materials used in electronic gadgets, recycling of e-plastic waste (waste to wealth).

Elastomers: -Introduction, vulcanization of rubber, advantages of vulcanized rubber, preparation, properties and applications (Buna S, Thiokol and polyurethanes).

Composite materials: Fiber reinforced plastics, conducting polymers, biodegradable and biomedical polymers.

Course Outcomes: At the end of this unit, the students will be able to Analyze **types of plastics, methods of fabrication, the different types of composite plastic materials and interpret the mechanism of conduction in conducting polymers.**

UNIT II: ELECTRO CHEMICAL CELLS AND CORROSION (**10hrs)**

Electrochemical cell, Single electrode potential, standard hydrogen electrode, calomel electrode, construction of glass electrode, electrochemical series and uses of series, galvanic series, differentiation between electrochemical series and galvanic series, batteries (Dry cell, Li ion battery and zinc air cells).**Corrosion**: -Definition, factors influencing rate of corrosion, theories of corrosion (direct chemical attack and electrochemical), galvanic corrosion, differential aeration corrosion, stress corrosion, corrosion control (proper designing and cathodic protection), Protective coatings (surface preparation, cathodic coatings, anodic coatings, electroplating and electroless plating [nickel]),Paints(constituents and special paints).

Course Outcomes:

At the end of this unit, the students will be able to Utilize the theory of construction of electrodes, batteries in redesigning new engineering products and categorize the reasons for corrosion and study methods to control corrosion**.**

UNIT III: MATERIAL CHEMISTRY **(10hrs)**

PART I:

NANOMATERIALS:

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

Super conductors: TypeI, TypeII, characteristics and applications

PART II:

Non–elemental semiconducting materials:

Stoichiometric, controlled valency& chalcogen photo / semiconductors-preparation of semi conducting (distillation, zone refining, czochralski crystal pulling, epitaxy, diffusion, ion implantation) – semiconducting devices (p-n junction diode as rectifier, junction transistor).

Magnetic materials: ferro and ferri magnetism – Hall effect and its applications.

Course outcomes: at the end of this unit, the students will be able to

- **Synthesis** nanomaterials for modern advances of engineering technology analyze the applications of superconductors.
- **Summarize** the preparation of semiconducting; analyze the applications of Hall effect.

UNIT IV: NON–CONVENTIONAL ENERGY SOURCES & FUEL CELLS (8hrs)

NON–CONVENTIONAL ENERGY SOURCES:

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

Fuel cells: Construction, Working and Applications (H2-O2, CH3OH-O2, phosphoric acid and molten carbonate). **Spectroscopic techniques:** Electromagnetic spectrum, UV laws of absorption, instrumentation, chromophores and auxochromes, intensity shifts and applications. FTIR instrumentation and applications.

Course outcomes: At the end of this unit, the students will be able to

- **Design** models for energy by different natural sources.
- **Analyze** the working of and applications of Fuel cells.

UNIT V: ADVANCED CONCEPTS / TOPICS IN CHEMISTRY (8 hrs)

Computational chemistry: Introduction to computational chemistry, molecular modelling and docking studies, **Molecular switches:** characteristics of molecular motors and machines, rotaxanes and catenanes as Artificial molecular machines, prototypes – linear motions in rotaxanes, an acid – base controlled molecular shuttle, a molecular elevator, an autonomous light -powered molecular motor.

Course out comes: At the end of this unit, the students will be able to

- **Obtain** the knowledge of computational chemistry and molecular machines.

Standard books:

1. P.C. Jainand M.Jain "**Engineering Chemistry**", 15/e, Dhanpat Rai & Sons, Delhi,

(Latest edition).
2. Shikha Agarwal, "**Engineering Chemistry**", Cambridge University Press, New Delhi, (2019).
3. Shashi Chawla, "**Engineering Chemistry**", Dhanpat Rai Publication Co. (Latest edition).

Reference Books:

1. K. Sesha Maheshwaramma and Mridula Chugh, "**Engineering Chemistry**", Pearson India Edn.
2. O.G. Palana, "**Engineering Chemistry**", Tata Mc. Graw Hill Education Private Limited, (2009).
3. CNR Rao and JM Honig (Eds) "**Preparation and characterization of materials**" Academic press, NewYork (latest edition).
4. B.S. Murthy, P. Shankar and others, "**Text book of Nano science and Nano technology**", University press (latest edition).

Contribution of Course Outcomes (COs) towards the achievement of programme outcomes (POs) (Strong – 3, Moderate – 2, Weak – 1)

CO – PO MAPPING:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	-	-	-	-	-	-	-	-	-	-	-	-
CO2	1	-	-	-	-	-	-	-	-	-	-	-
CO3	-	-	-	-	-	-	-	-	-	-	-	-
CO4	-	-	-	-	-	-	-	-	-	-	-	-
CO5	-	-	-	-	2	-	-	-	-	-	-	1

Year & Sem	I Year – II Semester					
Course Code	22UCS04	L	T	P	SS	C
Course Name	DATA STRUCTURES	3	1	0	0	3

Course Objectives:

The objective of the course is to

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

Course Outcomes:

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

CO1: Discuss various sorting & searching Techniques

CO2: Use linked structures in writing programs

CO3: Use Stacks and Queues in Writing Programs

CO4: Use Trees in writing programs and demonstrate different methods for traversing trees

CO5: Demonstrate Graphs and Graph Traversals.

UNIT I

Data Structures - Definition, Classification of Data Structures, Operations on Data Structures, Abstract Data Type (ADT), Preliminaries of algorithms. Time and Space complexity.

Searching - Linear search, Binary search, Fibonacci search.

Sorting- Insertion sort, Selection sort, Exchange (Bubble sort, quick sort), distribution (radixsort), merging (Merge sort) algorithms.

UNIT II

Linked List: Introduction, Single linked list, Representation of Linked list in memory, Operations on Single Linked list-Insertion, Deletion, Search and Traversal, Reversing Single Linked list, Applications on Single Linked list- Polynomial Expression Representation, Addition, Sparse Matrix Representation using Linked List, Advantages and Disadvantages of Single Linked list, Double Linked list-Insertion, Deletion, Circular Linked list-

Insertion, Deletion.

UNIT III

Stacks: Introduction to Stacks, Array Representation of Stacks, Operations on Stacks, linked list Representation of Stacks, Operations on Linked Stack, Applications-Reversing list, Factorial Calculation, Infix to Postfix Conversion, Evaluating Postfix Expressions.

Queues: Introduction to Queues, Representation of Queues-using Arrays and using Linked list, Implementation of Queues-using Arrays and using Linked list, Application of Queues- Circular Queues, Deques, Priority Queues.

UNIT IV

Trees: Basic Terminology in Trees, Binary Trees-Properties, Representation of Binary Trees using Arrays and Linked lists. Tree Traversal – In order, Preorder, Post order. Heaps –Definition, Max Heap, Min Heap, Insertion and Deletion from Max Heap, Heap sort, Binary Search Trees – Basic Concepts, BST Operations: Insertion and Deletion.

UNIT V

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

Text Books:

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

Reference Books:

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

e-Resources:

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

Contribution of Course Outcomes (COs) towards the achievement of programme

outcomes (POs) (Strong – 3, Moderate – 2, Weak – 1)

CO – PO MAPPING:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	3	2	3							2
CO2	3	1	2	2	3							1
CO3	3	1	2	2	3							2
CO4	3	3	2	2	3							2
CO5					1							2

Year & Sem	I Year – II Semester					
Course Code	22UCS05	L	T	P	SS	C
Course Name	PYTHON PROGRAMMING	3	1	0	0	3

Course Objectives:

The Objectives of Python Programming are

- To learn about Python programming language syntax, semantics, and the runtimeenvironment
- To be familiarized with universal computer programming concepts like data types, containers
- To be familiarized with general computer programming concepts like conditional execution,loops & functions
- To be familiarized with general coding techniques and object-oriented programming

Course Outcomes:

CO1: Develop essential programming skills in computer programming concepts like data types, containers
CO2: Apply the basics of programming in the Python language
CO3: Solve coding tasks related conditional execution, loops
CO4: Solve coding tasks related to the fundamental notions and techniques used in object-oriented programming
CO5: Design applications using GUI and Handle Exceptions

UNIT I

Introduction: Introduction to Python, Program Development Cycle, Input, Processing, and Output, Displaying Output with the Print Function, Comments, Variables, Reading Input from the Keyboard, Performing Calculations, Operators. Type conversions, Expressions, More about Data Output.
Data Types, and Expression: Strings Assignment, and Comment, Numeric Data Types and CharacterSets, Using functions and Modules.
Decision Structures and Boolean Logic: if, if-else, if-elif-else Statements, Nested Decision Structures, Comparing Strings, Logical Operators, Boolean Variables.
Repetition Structures: Introduction, while loop, for loop, Calculating a Running Total, Input Validation Loops, Nested Loops
UNIT II
Control Statement: Definite iteration for Loop Formatting Text for output, Selection if and if else Statement Conditional Iteration the While Loop

Strings and Text Files: Accessing Character and Substring in Strings, Data Encryption, Strings and Number Systems, String Methods Text Files.

UNIT III

List and Dictionaries: Lists, Defining Simple Functions, Dictionaries

Design with Function: Functions as Abstraction Mechanisms, Problem Solving with Top-Down Design, Design with Recursive Functions, Case Study Gathering Information from a File System, Managing a Program's Namespace, Higher Order Function.

Modules: Modules, Standard Modules, Packages.

UNIT IV

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

Object Oriented Programming: Concept of class, object and instances, Constructor, class attributes and destructors, Real time use of class in live projects, Inheritance, overlapping and overloading operators, Adding and retrieving dynamic attributes of classes, Programming using Oops support **Design with Classes:** Objects and Classes, Data modeling Examples, Case Study An ATM, Structuring Classes with Inheritance and Polymorphism

UNIT V

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

Graphical User Interfaces: The Behavior of Terminal Based Programs and GUI - Based, Programs, Coding Simple GUI-Based Programs, Other Useful GUI Resources.

Programming: Introduction to Programming Concepts with Scratch.

Text Books

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

Reference Books:

1) Introduction to Python Programming, Gowrishankar.S, Veena A, CRC Press.
2) Introduction to Programming Using Python, Y. Daniel Liang, Pearson.

Contribution of Course Outcomes (COs) towards the achievement of programme

outcomes (POs) (Strong – 3, Moderate – 2, Weak – 1)

CO – PO MAPPING:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1	1	2							1
CO2	2	2	1	1	2							1
CO3	2	2	1	1	2							1
CO4	3	2	1	1	2							1
CO5	2	2	1	1	2							1

Year & Sem	I Year – II Semester					
Course Code	22UEC01	L	T	P	SS	C
Course Name	DIGITAL LOGIC DESIGN	3	1	0	0	3

Course Learning Objectives: This course will enable the students to

- Study about the number systems, complements, signed binary numbers and binary codes.
- Study about Boolean algebra; illustrate map method for minimization of switching functions.
- Design combinational logic circuits like Adders, Subtractors, Decoders, and Encoders.
- Describe Latches and Flip-Flops
- Learn about counters and registers.

Course Outcomes:

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

CO1: Define different number representation and conversion from one radix to other.

CO2: Explain the minimization techniques using Boolean algebra and K-map method.

CO3: Analyze and design the combinational logic circuits.

CO4: Design the sequential circuits using Flip-Flops.

CO5: Design the registers and counters.

UNIT- I: Number Systems

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

UNIT -II: **Boolean algebra**

Basic Theorems and Properties of Boolean algebra, Boolean Functions, Canonical Forms, Minterms and Maxterms, Karnaugh Maps-3,4 variables, don't – Care terms, POS and SOP Simplification, NAND/ NOR Implementation, Logic gates.

UNIT –III: **Combinational Logic**

Analysis and design procedure for combinational logic, Adders and Subtractors, Binary Multiplier, Decoders, Encoders, Multiplexers, Demultiplexers, Priority Encoder, Code Converters, Magnitude Comparator.
Programmable logic devices: PROM, PAL, PLA

UNIT- IV: Synchronous Sequential Logic

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

UNIT -V: Registers and Counters

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

TEXT BOOKS:

1. Digital Design, M.Morris Mano, Michael D Ciletti, PEA.
2. Switching and finite automata theory Zvi.KOHAVI, Niraj.K.Jha 3rdEdition,Cambridge UniversityPress,2009
3. Switching theory and logic design by Hill and Peterson**,** Mc-Graw Hill TMH edition, 2012.

REFERENCE BOOKS:

1. Modern Digital Electronics, R.P. Jain, TMH.
2. Switching Theory and Logic Design by A. Anand Kumar, PHI Learning pvt ltd, 2016.
3. Digital fundamentals by Thomas L. Floyd, Pearson EDU India.

Contribution of Course Outcomes (COs) towards the achievement of programme outcomes (POs) (Strong – 3, Moderate – 2, Weak – 1)
CO – PO MAPPING:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	1	2	-	-	-	-	-	-	-	1
CO2	3	3	2	2	-	-	-	-	-	-	-	1
CO3	3	3	3	1	2	-	-	-	-	-	-	2
CO4	3	3	3	1	2	-	-	-	-	-	-	2
CO5	3	3	3	1	2	-	-	-	-	-	-	2

Year & Sem	I Year – II Semester					
Course Code	22UCH02	L	T	P	SS	C
Course Name	APPLIED CHEMISTRY LABORATORY	0	0	3	0	1.5

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

1. Determination of HCl using standard Na2CO3 solution.
2. Determination of alkalinity of a sample containing Na2CO3 and NaOH.
3. Determination of Mn+2 using standard oxalic acid solution.
4. Determination of ferrous iron using standard K2Cr2O7 solution.
5. Determination of Cu+2 using standard hypo solution.
6. Determination of temporary and permanent hardness of water using standard EDTA solution.
7. Determination of Fe+3 by a colorimetric method.
8. Determination of the concentration of acetic acid using sodium hydroxide (pH-metry method).
9. Determination of pH by using pH-meter.
10. Determination of the concentration of strong acid vs strong base (by conductometric method).
11. Determination of strong acid vs strong base (by potentiometric method).
12. Determination of Mg+2 present in an antacid.
13. Determination of CaCO3 present in an egg shell.
14. Estimation of Vitamin C.
15. Determination of phosphoric content in soft drinks.
16. Adsorption of acetic acid by charcoal.
17. Preparation of nylon-6, 6 and Bakelite (demonstration only).

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

Outcomes:

The students entering into the professional course have practically very little exposure to lab classes. The experiments introduce volumetric analysis; redox titrations with different indicators; EDTA titrations; then they are exposed to a

few instrumental methods of chemical analysis. Thus, at the end of the lab course, the student is exposed to different methods of chemical analysis and use of some commonly employed instruments. They thus acquire some experimental skills.

Reference Books

1. A Textbook of Quantitative Analysis, Arthur J. Vogel.

Contribution of Course Outcomes (COs) towards the achievement of programme outcomes (POs) (Strong – 3, Moderate – 2, Weak – 1)

CO – PO MAPPING:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	1	2	-	-	-	-	-	-	-	1
CO2	3	3	2	2	-	-	-	-	-	-	-	1
CO3	3	3	3	1	2	-	-	-	-	-	-	2
CO4	3	3	3	1	2	-	-	-	-	-	-	2
CO5	3	3	3	1	2	-	-	-	-	-	-	2

Year & Sem	I Year – II Semester					
Course Code	22UCS06	L	T	P	SS	C
Course Name	DATA STRUCTURES USING C LABORATORY	0	0	3	0	1.5

Course Objectives:

The objective of this lab is to

- Demonstrate the different data structures implementation.

Course Outcomes:

By the end of this lab the student is able to

CO1: Use various searching and sorting algorithms

CO2: Use basic data structures such as arrays and linked list.

CO3: Programs to demonstrate fundamental algorithmic problems including Stacks, Queues and Tree Traversals.

List of Experiments:

Exercise -1 (Searching)

a) Write C program that use both recursive and non-recursive functions to perform Linearsearch for a key value in a given list.

b) Write C program that use both recursive and non-recursive functions to perform Binarysearch for a key value in a given list.

Exercise -2 (Sorting-I)

a) Write C program that implement Bubble sort, to sort a given list of integers in ascendingorder

b) Write C program that implement Quick sort, to sort a given list of integers in ascendingorder

c) Write C program that implement Insertion sort, to sort a given list of integers in ascendingorder

Exercise -3(Sorting-II)

a) Write C program that implement radix sort, to sort a given list of integers in ascendingorder

b) Write C program that implement merge sort, to sort a given list of integers in ascendingorder

Exercise -4(Singly Linked List)

a) Write a C program that uses functions to create a singly linked list
b) Write a C program that uses functions to perform insertion operation on a singly linked list
c) Write a C program that uses functions to perform deletion operation on a singly linked list
d) Write a C program to reverse elements of a single linked list.

Exercise -5(Stack)

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

Exercise -6(Queue)

a) Write C program that implement Queue (its operations) using arrays.
b) Write C program that implement Queue (its operations) using linked lists

Exercise -7(Binary Tree)

d) Write a recursive C program for traversing a binary tree in preorder, in order and post order.

Exercise -8(Binary Search Tree)

a) Write a C program to Create a BST
b) Write a C program to insert a node into a BST.
c) Write a C program to delete a node from a BST.

Contribution of Course Outcomes (COs) towards the achievement of programme outcomes (POs) (Strong – 3, Moderate – 2, Weak – 1)

CO – PO MAPPING:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	3	2	3							2
CO2	3	1	2	2	3							2
CO3	3	1	2	2	3							2

Year & Sem	I Year – II Semester					
Course Code	22UCS07	L	T	P	SS	C
Course Name	PYTHON PROGRAMMING LABORATORY	0	0	3	0	1.5

Course Objectives:

The Objectives of Python Programming are

- To learn about Python programming language syntax, semantics, and the runtimeenvironment
- To be familiarized with universal computer programming concepts like data types,containers
- To be familiarized with general computer programming concepts like conditionalexecution, loops & functions
- To be familiarized with general coding techniques and object-oriented programming

Course Outcomes:

CO1: Develop essential programming skills in computer programming concepts like data types, containers

CO2: Apply the basics of programming in the Python language

CO3: Solve coding tasks related conditional execution, loops

CO4: Solve coding tasks related to the fundamental notions and techniques used in object-oriented programming

List of Experiments:

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

*

**

***

****

6) Generate a random number between 1 and 10. Ask the user to guess the number andprint a message based on whether they get it right or not.
7) Write a program that asks the user for two numbers and prints Close if the numbers arewithin .001 of each other and not close otherwise.
8) Write a program that asks the user to enter a word and prints out whether that word contains any vowels.
9) Write a program that asks the user to enter two strings of the same length. The program should then check to see if the strings are of the same length. If they are not the program should print an appropriate message and exit. If they are of the same length, the program should alternate the characters of the two strings. For example, if the user enters abcde and ABCDE the program should print out AaBbCcDdEe.
10) Write a program that asks the user for a large integer and inserts commas into it according to the standard American convention for commas in large numbers. For instance, if the user enters 1000000, the output should be 1,000,000.
11) In algebraic expressions, the symbol for multiplication is often left out, as in 3x+4y or 3(x+5). Computers prefer those expressions to include the multiplication symbol, like 3*x+4*y or 3*(x+5). Write a program that asks the user for an algebraic expression and then inserts multiplication symbols where appropriate.
11)Write a program that generates a list of 20 random numbers between 1 and 100.

(a) Print the list.
(b) Print the average of the elements in the list.
(c) Print the largest and smallest values in the list.
(d) Print the second largest and second smallest entries in the list
(e) Print how many even numbers are in the list.

12) Write a program that asks the user for an integer and creates a list that consists of the factors of that integer.
13)Write a program that generates 100 random integers that are either 0 or 1. Then find the longest run of zeros, the largest number of zeros in a row. For instance, the longest run of zeros in [1,0,1,1,0,0,0,0,1,0,0] is 4.
14)Write a program that removes any repeated items from a list so that each item appears at most once. For instance, the list [1,1,2,3,4,3,0,0] would become [1,2,3,4,0].
15)Write a program that asks the user to enter a length in feet. The program should then give the user the option to convert from feet into inches, yards, miles, millimeters, centimeters, meters, or kilometers. Say if the user enters a 1, then the program converts to inches, if they enter a 2, then the program converts to yards, etc. While this can be done with if statements, it is much shorter with lists and it is also easier to add new conversions if you use lists.
16)Write a function called sum digits that is given an integer num and returns the sum ofthe digits of num.

17)Write a function called first_diff that is given two strings and returns the first locationin which the strings differ. If the strings are identical, it should return -1.

18)Write a function called number_of_factors that takes an integer and returns how many factors the number has.

19)Write a function called is_sorted that is given a list and returns True if the list is sorted and False otherwise.

20)Write a function called root that is given a number x and an integer n and returns x1/n. In the function definition, set the default value of n to 2.

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

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

(a) Do this using the sort method. (b) Do this without using the sort method.

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

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

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

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

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

27)Write a class called Converter. The user will pass a length and a unit when declaring an object from the class—for example, c = Converter(9,'inches'). The possible units are inches, feet, yards, miles, kilometers, meters, centimeters, and millimeters. For each of these units there should be a method that returns the length converted into those units. For example, using the Converter object created above, the user could call c.feet() and shouldget 0.75 as the result.

28)Write a Python class to implement pow (x, n).
29)Write a Python class to reverse a string word by word.
30)Write a program that opens a file dialog that allows you to select a text file. The program then displays the contents of the file in a textbox.
31)Write a program to demonstrate Try/except/else.
32)Write a program to demonstrate try/finally and with/as.

Contribution of Course Outcomes (COs) towards the achievement of programme outcomes (POs) (Strong – 3, Moderate – 2, Weak – 1)

CO – PO MAPPING:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1	1	2							1
CO2	2	2	1	1	2							1
CO3	2	2	1	1	2							1
CO4	3	2	1	1	2							1

Year & Sem	I Year – II Semester					
Course Code	22UCH03	L	T	P	SS	C
Course Name	ENVIRONMENTAL SCIENCE	2	0	0	0	0

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

COURSE OBJECTIVES

The objectives of the course are to impart:

- Overall understanding of the natural resources.
- Basic understanding of the ecosystem and its diversity.
- Acquaintance on various environmental challenges induced due to unplanned Anthropogenic activities.
- An understanding of the environmental impact of developmental activities.
- Awareness on the social issues, environmental legislation and global treaties.

COURSE OUTCOMES

CO1: The natural resources and their importance for the sustenance of the life and recognize the need to conserve the natural resources

CO2: The concepts of the ecosystem and its function in the environment. The need for protecting the producers and consumers in various ecosystems and their role in the food web

CO3: The biodiversity of India and the threats to biodiversity, and conservation practices to protect the biodiversity

CO4: Various attributes of the pollution and their impacts and measures to reduce or control the pollution along with waste management practices

CO5: About environmental assessment and the stages involved in EIA and the environmental audit.

UNIT-I:

Multidisciplinary nature of Environmental Studies: Definition, Scope and Importance – Sustainability: Stockholm and Rio Summit–Global Environmental Challenges: Global warming and climate change, acid rains, ozone layer depletion, population growth and explosion, effects; Role of information technology in environment and human health.

Ecosystems: Concept of an ecosystem. - Structure and function of an ecosystem; Producers, consumers and decomposers. - Energy flow in the ecosystem - Ecological succession. - Food chains, food webs and ecological pyramids; Introduction, types, characteristic features, structure and function of Forest ecosystem, Grassland ecosystem, Desert ecosystem, Aquatic ecosystems.

UNIT-II:

Natural Resources: Natural resources and associated problems.

Forest resources: Use and over – exploitation, deforestation – Timber extraction – Mining, dams and other effects on forest and tribal people.
Water resources: Use and over utilization of surface and ground water – Floods, drought, conflicts over water, dams – benefits and problems.
Mineral resources: Use and exploitation, environmental effects of extracting and using mineral resources. Food resources: World food problems, changes caused by non-agriculture activities-effects of modern agriculture, fertilizer-pesticide problems, water logging, salinity. Energy resources: Growing energy needs, renewable and non-renewable energy sources use of alternate energy sources.
Land resources: Land as a resource, land degradation, Wasteland reclamation, man induced landslides, soil erosion and desertification; Role of an individual in conservation of natural resources; Equitable use of resources for sustainable lifestyles.

UNIT-III:

Biodiversity and its conservation: Definition: genetic, species and ecosystem diversity- classification - Value of biodiversity: consumptive use, productive use, social-Biodiversity at national and local levels. India as a mega-diversity nation - Hot-sports of biodiversity - Threats to biodiversity: habitat loss, man- wildlife conflicts. - Endangered and endemic species of India – Conservation of biodiversity: conservation of biodiversity.

UNIT – IV:

Environmental Pollution: Definition, Cause, effects and control measures of Air pollution, Water pollution, Soil pollution, Noise pollution, nuclear hazards. Role of an individual in prevention of pollution. - Pollution case studies, Sustainable Life Studies. Impact of Fire Crackers on Men and his wellbeing.
Solid Waste Management: Sources, Classification, effects and control measures of urban and industrial solid wastes. Consumerism and waste products, Biomedical, Hazardous and e– waste management.

UNIT – V:

Social Issues and the Environment: Urban problems related to energy -Water conservation, rain water harvesting - Resettlement and rehabilitation of people; its problems and concerns.

Environmental ethics: Issues and possible solutions. Environmental Protection Act - Air (Prevention and Control of Pollution) Act. –Water (Prevention and control of

Pollution) Act
-Wildlife Protection Act - Forest Conservation Act-Issues involved in enforcement of environmental legislation. -Public awareness. **Environmental Management**: Impact Assessment and its significance various stages of EIA, preparation of EMP and EIS, Environmental audit. Ecotourism, Green Campus – Green business and green politics. The student should Visit an Industry / Ecosystem and submit a report individually on any issues related to Environmental Studies course and make a power point presentation.

Text Books:

1. Environmental Studies, K. V. S. G. Murali Krishna, VGS Publishers,Vijayawada
2. Environmental Studies, R. Rajagopalan, 2nd Edition, 2011, Oxford UniversityPress.
3. EnvironmentalStudies,P.N.Palanisamy,P.Manikandan, A.Geetha,andK.Manjula Rani; PearsonEducation,Chennai

Reference:

1. TextBook of Environmental Studies, Deeshita Dave & P.UdayaBhaskar, CengageLearning.
2. A Textbook of Environmental Studies, Shaashi Chawla, TMH, NewDelhi
3. Environmental Studies, Benny Joseph, Tata McGraw Hill Co, NewDelhi
4. Perspectives in Environment Studies, Anubha Kaushik, C P Kaushik, NewAge InternationalPublishers,2014

Contribution of Course Outcomes (COs) towards the achievement of programme outcomes (POs) (Strong – 3, Moderate – 2, Weak – 1)

CO – PO MAPPING:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1			3			2	3					
CO2			3			2	3					
CO3			3			2	3					
CO4			3			2	3					
CO5			3			2	3					

Year & Sem	II Year – I Semester				
Course Code	22UMT03	L	T	P	C
		3	0	0	3
Course Name	MATHEMATICS-III (Partial Differential Equations and Vector Calculus)				

Course Objectives:

- To develop the students to solve real time engineering problems using partial differential equations.
- To familiarize the techniques in partial differential equations to describe a wide range of natural processes and other areas of mathematics such as analysis and differential geometry.
- To understand the properties of Beta and Gamma functions with their integral representations.
- To prepare the students to learn the concepts of Vector Calculus.
- To spread out the use of different numerical techniques for carrying out numerical integration.

Course Outcomes:

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

CO1: Solve problems related to basic linear and non-linear partial differential equations.

CO2: Identify solution methods for partial differential equations that model physical processes.

CO3: Explain the applications and the usefulness of the Beta and Gamma functions by their integral representations and symmetries.

CO4: Interpret the physical meaning of different operators such as gradient, curl, divergence and estimates the work done against a field, circulation and flux using vector calculus.

CO5: Apply Numerical Integration techniques to different engineering problems.

UNIT-I First Order PDE

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

[Text Book (1) : Sections – 17.2, 17.3, 17.5, 17.6]

UNIT II Higher Order Linear PDE and Applications

Solutions of linear partial differential equations with constant coefficients – non-homogeneous term of the type e^{ax+by}, $\sin(ax+by)$, $\cos(ax+by)$, $x^m y^n$. Applications of

PDE: Method of separation of Variables –Solution of One – dimensional Wave, Heat and two – dimensional Laplace equation.
[Text Book (1) : Sections – 17.8, 17.9, 17.10, 17.11, 18.2, 18.5, 18.6, 18.7]

UNIT III Beta and Gamma Functions

Beta and Gamma functions – Properties – Relation between Beta and Gamma functions – Evaluation of improper integrals.

[Text Book (1) : Sections – 7.14, 7.15, 7.16]

UNIT IV Vector Calculus

Vector Differentiation: Gradient – Directional derivative – Divergence –Curl – Scalar Potential Vector Integration: Line integral – Work done – Area– Surface and volume integrals – Vector integral theorems: Greens, Stokes and Gauss Divergence theorems (without proof) and problems on above theorems.

[Text Book (1) : Sections – 8.4, 8.5, 8.6, 8.7, 8.8, 8.9, 8.11, 8.12, 8.13, 8.14, 8.15, 8.16]

UNIT-V Numerical Integration and Solution of Ordinary Differential Equations

Numerical Integration - Trapezoidal rule– Simpson's $1/3^{rd}$ and $3/8^{th}$ rules – Solution of initial value problems by Taylor's series – Picard's method of successive approximations – Euler's method – Runge - Kutta method (fourth order only).
[Text Book (1) : Sections –30.4, 30.6, 30.7, 30.8, 32.2, 32.3, 32.4, 32.5, 32.6, 32.7,]

Text Books:

1. **B.S.Grewal,** Higher Engineering Mathematics, 44^{th} Edition, Khanna Publishers.

Reference Books:

1. **Dr.T.K.V.Iyengar, Dr. B. Krishna Gandhi, S. Ranganadham, Dr. M.V.S.S.N. Prasad**, A text book of Engineering Mathematics, S.Chand Publications.
2. **B.V.Ramana,** Higher Engineering Mathematics, 2007 Edition, Tata Mc.Graw Hill Education.
3. **Erwin Kreyszig,** Advanced Engineering Mathematics, 10^{th} Edition, Wiley - India.

Contribution of Course Outcomes (CO's) towards the achievement of programme outcomes (PO's) and Programme Specific Outcomes (PSO's)
(Strong – 3, Moderate – 2, Weak – 1)

CO – PO and PSO MAPPINGS

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	3	3	1	-	-	-	-	-	-	-	1	-	-	-
CO2	3	3	3	2	-	-	-	-	-	-	-	2	-	-	-
CO3	2	3	3	1	-	-	-	-	-	-	-	2	-	-	-
CO4	2	3	3	1	-	-	-	-	-	-	-	3	-	-	-
CO5	3	3	3	2	-	-	-	-	-	-	-	2	-	-	-

Year & Sem	II Year – I Semester				
Course Code	22UCS08	L	T	P	C
		3	0	0	3
Course Name	OBJECT ORIENTED PROGRAMMING THROUGH C++				

Course Objectives:

- Describe the procedural and object-oriented paradigm.
- Describe the object-oriented paradigm concepts of classes and objects, constructors, destructors
- Understand the operator overloading and classification of inheritance concepts.
- Understand the dynamic memory management techniques using pointers and dynamic binding.
- Demonstrate the use of exception handling, generic programming

Course Outcomes:

By the end of the course, the student

- Classify the object-oriented programming and procedural programming paradigm.
- Apply C++ features such as composition of objects, classes, constructors and destructors.
- Illustrate the operator overloading and inheritance concepts.
- Illustrate the dynamic memory management techniques using pointers and dynamic binding.
- Know about the use of exception handling, generic programming

UNIT I

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

UNIT II

Classes and Objects: Classes in C++, Declaring Objects, Access Specifiers and their Scope, Defining Member Function, Overloading Member Function, Nested class

Constructors and Destructors: Characteristics of Constructor and Destructor, Application with Constructor, Constructor with Arguments parameterized Constructor, Destructors, Anonymous Objects.

UNIT III

Operator Overloading and Type Conversion: The Keyword Operator, Overloading Unary Operator, Operator Return Type, Overloading Assignment Operator (=), Rules for Overloading Operators

Inheritance: Reusability, Types of Inheritance, Virtual Base Classes- Object as a Class Member, Abstract Classes, Advantages of Inheritance, Disadvantages of Inheritance.

UNIT IV

Pointers: Introduction, Features of Pointers, Pointer Declaration, Pointer to Class, Pointer Object, The this Pointer, Pointer to Derived Classes and Base Class

Binding Polymorphisms and Virtual Functions: Introduction, Binding in C++, Virtual Functions, Rules for Virtual Function, Virtual Destructor.

UNIT V

Generic Programming with Templates: Definition of class Templates, Normal Function Templates, Over Loading of Template Function, Difference between Templates and Macros

Exception Handling: Principles of Exception Handling, The Keywords try throw and catch, Multiple Catch Statements, Specifying Exceptions.

Text Books:

1) A First Book of C++, Gary Bronson, Cengage Learning.

2) The Complete Reference C++, Herbert Schildt, TMH.

Reference Books:

1) Object Oriented Programming C++, Joyce Farrell, Cengage.

2) C++ Programming: from problem analysis to program design, DS Malik, Cengage Learning

3) Programming in C++, Ashok N Kamthane, Pearson 2nd Edition

e- Resources:

1) https://nptel.ac.in/courses/106/105/106105151/

2) https://github.com/topics/object-oriented-programming

CO-PO MAPPING

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO.1	2	-	-	-	-	1	-	-	-	-	1	1
CO.2	2	2	3	2	-	2	-	-	2	-	2	2
CO.3	2	2	3	3	-	3	-	-	3	-	3	2
CO.4	2	2	3	3	-	3	-	-	3	-	3	2
CO.5	2	2	3	3	-	2	-	-	2	-	2	1

Year & Sem	II Year – I Semester				
Course Code	22UCS09	L	T	P	C
		3	0	0	3
Course Name	COMPUTER ORGANIZATION				

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Course Objectives:

- To understand the architecture of a modern computer with its various processing units. Also
  the performance measurement of the computer system.
- To understand the memory management system of computer.
- To understand the various instructions, addressing modes
- To understand the concept of I/O organization

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

- Understand the architecture of modern computer.
- Analyze the Performance of a computer using performance equation
- Understanding of different instruction types and addressing modes.
- Understand how computer stores positive and negative numbers.
- Understand the concepts of I/O Organization and Memory systems.

UNIT-I: Basic Structure of Computers: Computer Types, Functional unit, Basic Operational concepts, Bus structures, System Software, Performance, The history of computer development.
Machine Instruction and Programs: Instruction and Instruction Sequencing: Register Transfer Notation, Assembly Language Notation, Basic Instruction Types.

UNIT-II: Addressing Modes, Basic Input/output Operations, The role of Stacks and Queues in computer programming equation. **Component of Instructions:** Logic Instructions, Shift and Rotate Instructions.
Type of Instructions: Arithmetic and Logic Instructions, Branch Instructions, Addressing Modes, Input/output Operations.
UNIT-III: 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)
UNIT-IV: The Memory Systems: Basic Memory Circuits, Memory System Consideration, Read- Only Memory: ROM, PROM, EPROM, EEPROM, Flash

Memory, Cache Memories: Mapping Functions, Interleaving, **Secondary Storage:** Magnetic Hard Disks, Optical Disks

UNIT-V: 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.

TEXTBOOKS:

1. Computer Organization, Carl Hamacher, Zvonks Vranesic, SafeaZaky, 5thEdition, McGrawHill,2011.
2. Computer Organization and Architecture – William Stallings SixthEdition, Pearson/PHI

REFERENCE BOOKS:

1. Computer Architecture and Organization, John P. Hayes,3rdEdition, McGrawHill,2002.
2. Structured Computer Organization – Andrew S. Tanenbaum, 4th Edition PHI/Pearson, 2012.
3. Fundamentals or Computer Organization and Design, - Sivaraama Dandamudi Springer Int.Edition,2003.
4. "Computer Organization and Design: The Hardware/Software Interface" by David A. Patterson and John L. Hennessy, 1998.

CO-PO MAPPING

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO.1	3	2	-	-	-	-	-	-	-	1	-	1
CO.2	3	2	2	1	-	-	-	-	1	1	-	1
CO.3	3	2	2	1	-	-	-	-	1	1	-	1
CO.4	2	2	2	2	-	-	-	-	1	1	-	1
CO.5	3	2	2	1	-	-	-	-	1	1	-	1

Year & Sem	II Year – I Semester				
Course Code	22UCS10	L	T	P	C
		3	0	0	3
Course Name	SOFTWARE ENGINEERING				

Course Objectives

- The aim of the course is to provide an understanding of the working knowledge of the techniques for estimation, analysis, design, testing large software development projects.
- Topics include introduction concepts of software and software engineering, process models, agile development, estimation, software requirements, software design, software testing and UML diagrams.

Course Outcomes

After completion on the course, Students will gain software engineering skills in the following areas:

- To know about Software and Software Engineering process
- Knowledge on various Conventional and Agile process models
- Able to translate end-user requirements into system and software requirements and structure the requirements in a Software Requirements Specification Document (SRSD).
- Able to build models of the system using appropriate models like DFDs, UML Diagrams etc.
- Skills to design, implement and execute test case to the test the software product.

UNIT - I

Software and Software Engineering: Nature of software, Unique nature of WebApps, Software engineering- a layered technology, Software myths, Software Engineering Practice, The Software process, Software Myths. (Text Book -1)

A Generic Process Model: Defining Frame Work Activity, Identifying a Task Set, Process Patterns, Process Assessment and Improvement. (Text Book -1)

UNIT-II

Prescriptive Process models: The Waterfall model, Incremental process models, Evolutionary process models, Specialized process model, the Unified process, Personal and Team process models. (Text Book -1)

Agile Development: Agility, Agility and the cost of change, Agile Process Models-Extreme Programming (XP), Scrum, Agile Unified Process (AUP) (Text Book -1)

UNIT - III

Requirements Engineering: Functional and non-functional requirements, the Software requirements document, Requirement specification, Requirements Engineering processes, Requirements elicitation and analysis, Requirements validation, Requirements management (Text Book -2)

Estimation of Software Projects: FP based Estimation, LOC based Estimation, Empirical Estimation Models-COCOMO (Text Book-1)

UNIT-IV

Modeling Basics: DFD, Structured Charts, E-R diagrams, UML diagrams
(Text Book -3)

Requirement Modeling: Scenario-Based Modeling, Class-Based Modeling, Flow-Oriented Modeling, Creating a Behavioral Model, Data Modeling (Text Book -1)

UNIT V

Design Engineering and Modeling: Design concepts, Design model-Data Design, Architectural Design, Interface Design, Component Level Design, Architectural Styles and Patterns (Text Book -1)

Software Testing: A strategic approach to software testing, test strategies for conventional software, test strategies for OO software, system testing, validation testing, the art of debugging, White box and Black box testing. (Text Book -1)

TEXT BOOKS:

1. Software Engineering, A practitioner's Approach- Roger S. Pressman, 6th edition, Mc Graw Hill International Edition.
2. Software Engineering- Ian Sommerville, 9th edition, Pearson Education.
3. The unified modeling language user guide Grady Booch, James Rambaugh, Ivar Jacobson, Pearson Education.

REFERENCE BOOKS:

1. Software Engineering, an Engineering approach- James F. Peters, Witold Pedrycz, John Wiley.
2. Software Engineering principles and practice- Waman S Jawadekar, The Mc Graw-Hill Companies.
3. Fundamentals of object-oriented design using UML Meiler page-Jones: Pearson Education.

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	-	-	-	-	1	-	-	-	-	-	-
CO2	1	2	3	-	-	1	-	-	3	3	3	2
CO3	1	2	3	3	-	1	-	-	3	3	3	2
CO4	1	2	3	3	3	1	-	-	3	3	3	2
CO5	1	2	2	3	3	1	-	-	3	3	3	2

Year & Sem	II Year – I Semester				
Course Code	22UCS11	L	T	P	C
		3	0	0	3
Course Name	MATHEMATICAL FOUNDATIONS OF COMPUTER SCIENCE				

Course Objectives:

This course is designed to:

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

Course Outcomes:

At the end of the course student will be able to

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

UNIT-I

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

UNIT-II

Relations: Relations: Properties, Operations, Partition and Covering, Transitive Closure, Equivalence, Compatibility and Partial Ordering, Hasse Diagrams,

Algebraic Structures: Algebraic Systems, Properties, Semi Groups and Monoids, Group, Subgroup and Abelian Groups - Homomorphism, Isomorphism.

UNIT-III

Combinatorics: Basis of Counting, Permutations, Permutations with Repetitions, Circular Permutations, Combinations. **Number Theory**: Properties of Integers, Division Theorem, Greatest Common Divisor, Euclidean Algorithm, Least Common Multiple, Testing for Prime Numbers, The Fundamental Theorem of Arithmetic, Modular Arithmetic, Fermat's and Euler's Theorems.

UNIT-IV

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

UNIT-V

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

Text Books:

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

Reference Books :

1. Discrete Mathematics for Computer Scientists and Mathematicians, .L. Mott, A.Kandel and T.P. Baker, 2nd Edition, Prentice Hall of india.

2. Discrete Mathematical Structures, Bernand Kolman, Robert C. Busby and Sharon Cutler Ross, PHI.
3. Discrete Mathematics and its Applications with Combinatories and Graph Theory, K.H.Rosen, 7th Edition, Tata McGraw Hill.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	1	3	1	-	-	-	-	-	-	-	-	-
CO2	2	2	-	-	-	-	-	-	-	-	-	-
CO3	2	3	1	-	2	-	-	-	-	-	-	-
CO4	2	2	-	2	-	1	-	-	-	-	-	-
CO5	1	2	-	-	-	-	-	-	-	-	-	-

(AUTONOMOUS)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Year & Sem	II Year – I Semester				
Course Code	22UCS12	L	T	P	C
		0	0	3	1.5
Course Name	OBJECT ORIENTED PROGRAMMING THROUGH C++ LAB				

Course Objectives:

The objective of this lab is to

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

Course Outcomes:

By the end of this lab the student is able to

- Apply C++ features such as composition of objects, classes, constructors and destructors.
- Apply the Inheritance concept with the help of sample programs
- Apply the polymorphism concept with the help of sample programs.
- Apply the Exception handling and generic programming with the help of sample programs.

Exercise -1 (Classes Objects)

1. Write a main function to create a class to add the given numbers.
2. Write a program to illustrate this pointer
3. Write a program for illustrating function overloading.

Exercise – 2 (Access)

1. Write a program implementing Friend Function
2. Write a C++ Program to illustrate the use of Constructors and Destructors
3. Write a Program to illustrate pointer to a class

Exercise -3 (Operator Overloading)

1. Write a program to Overload Unary, and Binary Operators as Member Function, and Non Member Function.
   i. Unary operator as member function
   ii. Binary operator as non member function

2. Write a c ++ program to implement the overloading assignment = operator

Exercise -4 (Inheritance)

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

Exercise -5(Templates, Exception Handling)

1. Write a C++ Program to illustrate template class
2. Write a Program to illustrate template function.
3. Write a Program for Exception Handling Divide by zero
4. Write a Program to rethrow an Exception

Exercise -6

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

CO-PO MAPPING

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO.1	3	-	-	-	-	2	-	-	-	-	1	1
CO.2	2	2	3	3	-	3	-	-	3	-	3	2
CO.3	2	2	3	3	-	3	-	-	3	-	3	2
CO.4	2	2	3	3	-	3	-	-	3	-	3	2

Year & Sem	II Year – I Semester				
Course Code	22UCS13	L	T	P	C
		0	1	2	2
Course Name	R PROGRAMMING LAB				

COURSE OBJECTIVES:

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

COURSE OUTCOMES:

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

- Access online resources for R and import new function packages into the R workspace
- Import, review, manipulate and summarize data-sets in R
- Explore data-sets to create testable hypotheses and identify appropriate statistical tests
- Perform appropriate statistical tests using R
- Create and edit visualizations with R

1. Download and install R-Programming environment and install basic packages.
2. Write a R program to take input from the user (name and age) and display the values. Also print the version of R installation.
3. Write a R program to get the details of the objects in memory.
4. Write a R program to create a sequence of numbers from 20 to 50 and find the mean of numbers from 20 to 60 and sum of numbers from 51 to 91.
5. Write a R program to create a simple bar plot of five subjects' marks.
6. Write a R program to get the unique elements of a given string and unique numbers of vector.

7. Write a R program to create three vectors a,b,c with 3 integers. Combine the three vectors to become a 3x3 matrix where each column represents a vector. Print the content of the matrix.
8. Write a R program to create a 5 x 4 matrix, 3 x 3 matrix with labels and fill the matrix by rows and 2 x 2 matrix with labels and fill the matrix by columns.
9. Write a R program to combine three arrays so that the first row of the first array is followed by the first row of the second array and then first row of the third array.
10. Write a R program to create a two-dimensional 5x3 array of sequence of even integers greater than 50.
11. Write a R program to create an array using four given columns, three given rows, and two given tables and display the content of the array.
12. Write a R program to create an empty data frame.
13. Write a R program to create a data frame from four given vectors.
14. Write a R program to create a data frame using two given vectors and display the duplicated elements and unique rows of the said data frame.
15. Write a R program to save the information of a data frame in a file and display the information of the file.
16. Write a R program to create a matrix from a list of given vectors.
17. Write a R program to concatenate two given matrices of same column but different rows.
18. Write a R program to find row and column index of maximum and minimum value in a given matrix.
19. Write a R program to append value to a given empty vector.
20. Write a R program to multiply two vectors of integers type and length 3.
21. Write a R program to find Sum, Mean and Product of a Vector, ignore element like NA or NaN.
22. Write a R program to list containing a vector, a matrix and a list and give names to the elements in the list.
23. Write a R program to create a list containing a vector, a matrix and a list and give names to the elements in the list. Access the first and second element of the list.
24. Write a R program to select second element of a given nested list.
25. Write a R program to merge two given lists into one list.
26. Write a R program to create a list named s containing sequence of 15 capital letters, starting from 'E',
27. Write a R program to assign new names "a", "b" and "c" to the elements of a given list.

28. Write a R program to find the levels of factor of a given vector.
29. Write a R program to create an ordered factor from data consisting of the names of months.
30. Write a R program to concatenate two given factors in a single factor.

CO-PO MAPPING

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	2	2	-	-	-	-	-	-	-	-	-
CO2	2	1	-	-	2	-	-	-	-	-	-	-
CO3	3	2	-	3	-	-	-	-	-	-	-	-
CO4	3	2	3	2	2	-	-	-	-	-	-	-
CO5	-	2	3	-	-	-	-	-	-	-	-	-

Year & Sem	II Year – I Semester				
Course Code	22UCS14	L	T	P	C
		0	0	2	1
Course Name	SOFTWARE ENGINEERING LAB				

Course Objectives:

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

Course Outcomes:

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

Consider any one application

- Course Registration System
- Student Attendance Management System
- Online Ticket Reservation System
- Library Management System
- Student Marks Analysing System
- ATM System
- Employee Management System.

1. Do the Requirement Analysis and Prepare SRS for it.

2. Draw E-R diagram for it

3. Draw DFD and Structured charts for it

4. Consider any application, using FP based Estimation to estimate effort

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

6. Draw the UML Diagrams for it

7. Design the test cases for it

8. Implement code for it.

9. Test the application by applying testing techniques (manual)

10. Use testing tool to test the application.

CO-PO MAPPING

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	1	2	3	3	-	1	-	-	3	3	3	2
CO2	1	2	3	3	_	1	-	-	3	3	3	2
CO3	1	2	3	3	3	1	-	-	3	3	3	2
CO4	1	2	3	3	3	1	-	-	3	3	3	2
CO5	1	2	3	3	3	1	-	-	3	3	3	2

Year & Sem	II Year – I Semester				
Course Code	22UCS15	L	T	P	C
		2	0	0	2
Course Name	Skill Oriented Course- I APPLICATIONS OF PYTHON-NumPy & PANDAS				

Course Objectives:

1. The objective of this lab is to acquire programming skills in Python package NumPy and perform mathematical and statistical operations.
2. The objective of this lab is to understand the fundamentals of the Pandas library in Python and how it is used to handle data and also develop basic skills in data analysis and visualization

Course Outcomes:

By the end of this lab the student is able to

- Determine Collection, managing of data and stored for processing.
- Apply some linear algebraic operations to n-dimensional arrays.
- Able to solve the engineering problems using Statistical & Mathematical Operations.
- Able to create and manipulate the data structures like Series & Data frames.
- Able to create basic charts & Histograms using Matplotlib function.

NUMPY

Perform the following:

1) NumPy Installation using different scientific python distributions (Anaconda, Python(x,y), WinPython, Pyzo)
2) NumPy Basics (np.array, np.arrange, np.linespace, np.zeros, np.ones, np.random.random, np.empty)
3) Arrays (array.shape, len(array), array.ndim, array.dtype, array.astype(type), type(array))
4) Array Manipulation (np.append, np.insert, np.resize, np.delete, np.concatenate, np.vstack, np.hstack)

5) Mathematical Operations(np.add, np.substract, np.divide, np.multiply, np.sqrt, np.sin, np.cos, np.log, np.dot, np.roots) , Statistical Operations(np.mean, np.median, np.std, array.corrcoef( ))

6) NumPy data types

7) NumPy ndarray

8) NumPy String Operations

9) NumPy Financial functions

10) NumPy Functional Programming

PANDAS

Perform the following:

1) Pandas Installation

2) Creating DataFrames

Exercises:

A) Pandas DataSeries:

1) Write a Pandas program to create and display a one-dimensional array-like object containing an array of data using Pandas module.

2) Write a Pandas program to convert a Panda module Series to Python list and it's type.

3) Write a Pandas program to add, subtract, multiple and divide two Pandas Series.

4) Write a Pandas program to convert a NumPy array to a Pandas series.

Sample Series: NumPy array:

[10 20 30 40 50] Converted Pandas series:

0 10

1 20

2 30

3 40

4 50

dtype: int64

B) Pandas DataFrames:

Consider Sample Python dictionary data and list labels:

exam_data = {'name': ['Anastasia', 'Dima', 'Katherine', 'James', 'Emily', 'Michael',

'Matthew', 'Laura', 'Kevin', 'Jonas'],

'score': [12.5, 9, 16.5, np.nan, 9, 20, 14.5, np.nan, 8, 19],

'attempts': [1, 3, 2, 3, 2, 3, 1, 1, 2, 1],

'qualify': ['yes', 'no', 'yes', 'no', 'no', 'yes', 'yes', 'no', 'no', 'yes']}

labels = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j']

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

2) Write a Pandas program to change the name 'James' to 'Suresh' in name column of the Data Frame.

3) Write a Pandas program to insert a new column in existing Data Frame.

4) Write a Pandas program to get list from Data Frame column headers.

5) Write a Pandas program to get list from Data Frame column headers. C) Pandas Index:

1) Write a Pandas program to display the default index and set a column as an Index in a given data frame.

2) Write a Pandas program to create an index labels by using 64-bit integers, using floating-point numbers in a given data frame.

D) Pandas String and Regular Expressions:

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

E) Pandas Joining and merging Data Frame:

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

2) Write a Pandas program to append a list of dictioneries or series to a existing DataFrame and display the combined data.

3) Write a Pandas program to join the two dataframes with matching records from both sides where available.

F) Pandas Time Series:

1) Write a Pandas program to create

   a) Datetime object for Jan 15 2012.

   b) Specific date and time of 9:20 pm.

   c) Local date and time.

   d) A date without time.

   e) Current date.

   f) Time from a datetime.

   g) Current local time.

2) Write a Pandas program to create a date from a given year, month, day and another date from a given string formats.

3) Write a Pandas program to create a time-series with two index labels and random values. Also print the type of the index.

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

G) Pandas Grouping Aggregate: Consider dataset:

1) Write a Pandas program to split the following dataframe into groups based on school code. Also check the type of GroupBy object.

2) Write a Pandas program to split the following dataframe by school code and get mean, min, and max value of age for each school.

H) Pandas Styling:

1) Create a dataframe of ten rows, four columns with random values. Write a Pandas program to highlight the negative numbers red and positive numbers black.

2) Create a dataframe of ten rows, four columns with random values. Write a Pandas program to highlight the maximum value in each column.

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

I) Excel:

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

J) Plotting:

1) Write a Pandas program to create a horizontal stacked bar plot of opening, closing stock prices of any stock dataset between two specific dates.

2) Write a Pandas program to create a histograms plot of opening, closing, high, low stock prices of stock dataset between two specific dates.

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

K) Pandas SQL Query:

1) Write a Pandas program to display all the records of a student file.

2) Write a Pandas

Mapping of Course Outcomes with Program Outcomes:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	3	3	3	3	3	3	-	-	1	1	2
CO2	3	3			_	1	-	-	1	1	1	1
CO3	1	2	3	3	3	1	-	-	3	3	3	2
CO4	3	3	3				-	-	3	3	3	2
CO5	2	3	3				-	-	3	3	3	2

ST. ANN'S COLLEGE OF ENGINEERING & TECHNOLOGY: CHIRALA

(AUTONOMOUS)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Year & Sem	II Year – I Semester				
Course Code	22UEN04	L	T	P	C
		2	0	0	0
Course Name	ESSENCE OF INDIAN TRADITIONAL KNOWLEDGE				

Course Objectives:

1. The course aims at imparting basic principles of thought process, reasoning and inferencing. Sustainability is at the core of Indian Traditional Knowledge Systems connecting society and nature.
2. Holistic life style of Yogic-science and wisdom capsules in Sanskrit literature are also important in modern society with rapid technological advancements and societal disruptions.
3. The course focuses on introduction to Indian Knowledge System, Indian perspective of modern scientific world- view and basic principles of Yoga and holistic health care system

Course Outcomes:

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

1. Understand the significance of Indian Traditional Knowledge
2. Classify the Indian Traditional Knowledge
3. Compare Modern Science with Indian Traditional Knowledge system.
4. Analyze the role of Government in protecting the Traditional Knowledge
5. Understand the impact of Philosophical tradition on Indian Knowledge System.

Unit I

Introduction to Traditional Knowledge: Define Traditional Knowledge- Nature and Characteristics- Scope and Importance- kinds of Traditional Knowledge- The historical impact of social change on Traditional Knowledge Systems- Value of Traditional knowledge in global economy.

Unit II

Basic structure of Indian Knowledge System: Astadash Vidya- 4 Ved - 4 Upaved (Ayurved, Dhanurved, Gandharva Ved & Sthapthya Adi), 6 vedanga (Shisha, Kalppa, Nirukha,Vykaran, Jyothisha & Chand),4 upanga (Dharmashastra, Meemamsa, purana & Tharka Shastra).

Unit III

Modern Science and Indian Knowledge System-Indigenous Knowledge, Characteristics- Yoga and Holistic Health care-cases studies.

Unit IV

Protection of Traditional Knowledge: The need for protecting traditional knowledge -Significance of Traditional knowledge Protection-Role of government to harness Traditional Knowledge.

Unit V

Impact of Traditions: Philosophical Tradition (Sarvadarshan) Nyaya, Vyshepec, Sankhya, Yog, Meemamsa, Vedantha, Chavanka, Jain & Boudh - Indian Artistic Tradition - Chitrakala, Moorthikala,Vasthukala , Sthapthya, Sangeetha, NruthyaYevamSahithya

Reference Books :

1. Traditional Knowledge System in India, by AmitJha, 2009.
2. Traditional Knowledge System and Technology in India by Basanta Kumar Mohanta andVipin Kumar Singh, PratibhaPrakashan 2012.
3. Sivaramakrishnan (Ed.), Cultural Heritage of India-course material, BharatiyaVidya
4. Swami Jitatmanand, Holistic Science and Vedant, BharatiyaVidyaBhavan
5. Yoga Sutra of Patanjali, Ramakrishna Mission, Kolkata.
6. Pramod Chandra, India Arts, Howard Univ. Press, 1983.
7. Krishna Chaitanya, Arts of India, Abhinav Publications, 1987.

Web Resources:

1. https://www.wipo.int/wipo_magazine/en/2017/01/article_0004.html
2. http://iks.iitgn.ac.in/wp-content/uploads/2016/01/Indian-Knowledge-Systems-Kapil-Kapoor.pdf
3. https://www.wipo.int/edocs/mdocs/tk/en/wipo_grtkf_ic_21/wipo_grtkf_ic_21_ref_facilitators_text.pdf

Mapping of Course Outcome

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	-											
CO2												
CO3												
CO4												
CO5								1				
CO6												

T. ANN'S COLLEGE OF ENGINEERING & TECHNOLOGY: CHIRALA

(AUTONOMOUS)

Year & Sem	II Year – II Semester				
Course Code	22UCS16	L	T	P	C
		3	0	0	3
Course Name	Probability and Statistics				

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

1. Course Objectives:

- To impart understanding of fundamental knowledge in probability and statistical techniques for gathering, organizing, displaying, and analyzing the significant data.
- To infuse the importance of random variables and distributions in decision making in order to carry out the necessary random experiments and identify the potential outcomes in a specific instance
- To provide the knowledge of Fitting the Appropriate Curves, Correlation and Regression Analysis to the data and estimate the future values.
- To enable the students for comprehending the Sampling Theory to estimate unknown population parameters using sample studies with the aid of statistical methodologies
- To analyze the concepts of Sampling, tests based on hypothesis to make an inference about the population of interest on the basis of a Random sample taken from the population.

Course Outcomes:

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

CO1: Demonstrate the basic knowledge on fundamental probability concepts, and various statistical methods of collecting data to interpret.

CO2: Understand the basic principles of distribution theory, mathematical expectation and various applications.

CO3: Interpret the association of Characteristics and through Correlation and Regression tools.

CO4: Estimate the confidence intervals for the mean of a population and Test a Hypothesis concerning means.

CO5: Design the components of a classical Hypothesis tests and Derive the

statistical inferential methods based on Small and Large Sampling tests.

UNIT-I Probability and Introductory Statistical Methods (12 Hours)

Probability – Probability Axioms – Addition Law and Multiplicative Law of Probability - Conditional probability - Baye's theorem. Measures of Central Tendency – Measures of Dispersion – Skewness – Kurtosis.

[Text Book (2): Sections- 4.5, 4.6, 4.6.2, 4.7, 4.8, 3.3, 3.13, 3.14,]

UNIT-II Random Variables and Probability Distributions (12 Hours)

Random variables (Discrete and continuous) – Probability Density Function - Probability Distribution function - Binomial, Poisson and Normal distribution – Related properties.

[Text Book (1) : Sections - 2.2, 2.3, 2.4, 2.4.1, 2.4.2, 2.5, 2.6, 2.7, 3.4,3.7, 3.10]

UNIT-III Correlation, Regression (14 Hours)

Correlation – Correlation Coefficient – Rank Correlation – Regression - Lines of Regression – Regression Coefficients – Method of Least Squares - Straight Line - Parabola - Exponential - Power Curves. [Text Book (1): Sections – 10.2, 10.8, 10.9, 10.10, 10.13, 11.1, 11.11, 11.15, 9.1, 9.2, 9.3]

UNIT-IV Sampling Theory and Estimation (14 Hours)

Sampling Theory Introduction – Population and Samples – Sampling Distribution of Means and Variance (Definitions Only) – Central Limit Theorem (Without Proof) - Introduction to t, χ^2 Distributions. Estimation: Point Estimation - Interval Estimation. [Text Book (1): Sections – 5.1, 5.2, 5.3, 5.4, 5.5, 5.6, 5.7, 5.8, 5.9, 5.10, 5.11, 6.2, 6.4, 6.5, 8.2, 8.3, 8.4, 8.5, 8.6]

UNIT-V Tests of Hypothesis (14 Hours)

Tests of Hypothesis Introduction – Hypothesis - Null Hypothesis - Alternative Hypothesis – Type-I and Type-II Errors - Level of Significance – One Tail and Two Tail Tests – Tests concerning one mean and two means (Large and Small Samples) – Tests on Proportions.

2. [Text Book(1): Sections – 7.2, 7.3, 7.4, 7.5, 7.6, 7.7, 7.8, 7.9, 8.8, 8.9, 8.11, 8.12]

3. **Text Books:**

1. Probability and Statistics by Dr.T.K.V.Iyengar, Dr. B.Krishna Gandhi, S.Ranganadham, Dr. M.V.S.S.N. Prasad, S.Chand Publications.
2. Fundamentals of Mathematical Statistics, S.C.Gupta and V.K.Kapoor, Sultan

Chand Publications, Tenth revised edition, 2002.

4. Reference Books:

1. Probability and Statistics for Engineers**,** Miller and John E. Freund, Prentice Hall of India, Eighth Edition, 2011.
2. Probability and Statistics (Schaum's Outline Series) by Murray Spiegel, John Schiller, McGraw Hill Education, 2005.
3. Probability, Statistics and Random processes. T. Veerrajan, Tata Mc.Graw Hill, India, Third Edition, 2009.

Contribution of Course Outcomes (CO) towards the achievement of programme outcomes (PO)

(Strong – 3, Moderate – 2, Weak – 1)

CO – POMAPPINGS

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	3	2	2	-	-	-	-	-	-	2
CO2	3	3	3	2	2	-	-	-	-	-	-	2
CO3	3	3	3	2	2	-	-	-	-	-	-	2
CO4	3	3	3	3	3	-	-	-	-	-	-	1
CO5	3	3	3	3	2	-	-	-	-	-	-	1

ST. ANN'S COLLEGE OF ENGINEERING & TECHNOLOGY: CHIRALA

(AUTONOMOUS)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Year & Sem	II Year – II Semester				
Course Code	22UCS17	L	T	P	C
		3	0	0	3
Course Name	OPERATING SYSTEMS				

Course Objectives:

The objectives of this course is to

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

Course Outcomes:

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

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

UNIT I

Operating Systems Overview: Operating system functions, Operating system structure, Operating systems operations, Computing environments, Open-Source Operating Systems. System Structures: Operating System Services, User and

Operating-System Interface, systems calls, Types of System Calls, system programs, operating system structure, operating system debugging, System Boot.

UNIT II

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

Communication in client server systems.

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

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

UNIT III

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

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

UNIT IV

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

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

UNIT V

File Systems: Files, Directories, File system implementation, management and optimization. Secondary-Storage Structure: Overview of disk structure, and attachment, Disk scheduling, RAID structure, Stable storage implementation.

System Protection: Goals of protection, Principles and domain of protection, Access matrix,

Access control, Revocation of access rights.

Case Studies: Linux, Microsoft Windows.

Text Books:

1) Silberschatz A, Galvin P B, and Gagne G, Operating System Concepts, 9th edition,
Wiley, 2013.

2) Tanenbaum A S, Modern Operating Systems, 3rd edition, Pearson Education, 2008. (for Interprocess Communication and File systems.)

Reference Books:

1) Dhamdhere D M, Operating Systems A Concept Based Approach, 3rd edition, Tata
McGraw-Hill, 2012.

2) Stallings W, Operating Systems -Internals and Design Principles, 6th edition, Pearson
Education, 2009

3) Nutt G, Operating Systems, 3rd edition, Pearson Education, 2004.

e-Resources:

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

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO.1	**3**	**3**	**3**	**3**	**3**	**2**	**1**	**1**	**1**	**1**	**1**	**3**
CO.2	**3**	**3**	**2**	**3**	**2**	**2**	**2**	**1**	**1**	**1**	**1**	**3**
CO.3	**3**	**3**	**3**	**3**	**3**	**3**	**1**	**1**	**1**	**1**	**1**	**3**
CO.4	**3**	**3**	**2**	**3**	**2**	**3**	**3**	**2**	**1**	**1**	**2**	**3**
CO.5	**3**	**2**	**2**	**2**	**2**	**3**	**3**	**2**	**1**	**1**	**2**	**3**

(AUTONOMOUS)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Year & Sem	II Year – II Semester				
Course Code	22UCS18	L	T	P	C
		3	0	0	3
Course Name	DATABASE MANAGEMENT SYSTEMS				

Course Objectives:

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

Course Outcomes:

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

- Describe a Database and Database schema
- Create, maintain and manipulate a relational database using SQL
- Describe ER model for database design and learn and use SQL sub queries, nested queries, aggregate functions, group by, order by, joins and views etc.
- Describe and learn normalization for database design
- Know about the transactions, concurrent access and Indexes to search data in the relation

UNIT I

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

UNIT II

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

UNIT III

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

UNIT IV

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

UNIT V

Transaction Concept: Transaction State, Implementation of Atomicity and Durability, Concurrent Executions, Serializability, Recoverability, Implementation of Isolation, Testing for Serializability, Failure Classification, Storage, Recovery and Atomicity, Recovery algorithm.

Text Books:

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

Reference Books:

1) Introduction to Database Systems, 8/e C J Date, PEA.

2) Database Management System, 6/e Ramez Elmasri, Shamkant B. Navathe, PEA
3) Database Principles Fundamentals of Design Implementation and Management, Corlos Coronel, Steven Morris, Peter Robb, Cengage Learning.

e-Resources:

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

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	-	-	-	-	1	-	-	-	-	-	-
CO2	1	2	3	-	-	1	-	-	-	-	-	1
CO3	1	2	3	3	-	1	-	-	1	1	1	1
CO4	1	2	3	3	-	1	-	-	1	1	1	1
CO5	1	2	2	3	-	1	-	-	1	1	1	1

ST. ANN'S COLLEGE OF ENGINEERING & TECHNOLOGY: CHIRALA

(AUTONOMOUS)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Year & Sem	II Year – II Semester				
Course Code	22UCS19	L	T	P	C
		3	0	0	3
Course Name	JAVA PROGRAMMING				

Course Objectives:

The learning objectives of this course are:

- To identify Java language components and how they work together in applications
- To learn the fundamentals of object-oriented programming in Java, including defining classes,

invoking methods, using class libraries.

- To learn how to extend Java classes with inheritance and dynamic binding and how to use

exception handling in Java applications

- To understand how to design applications with threads in Java
- To understand how to use Java APIs for program development

Course Outcomes:

By the end of the course, the student will be

- Able to realize the concept of Object Oriented Programming & Java Programming Constructs

- Able to describe the basic concepts of Java such as operators, classes, objects, inheritance,

  packages, Enumeration and various keywords

- Apply the concept of exception handling and Input/ Output operations

- Able to design the applications of Java & Java applet

- Able to design the applications of Event handling & AWT

UNIT-I:

Data Types, Variables, and Operators: Introduction to Java, Declaration of Variables , Data Types in Java, Keywords, Literals, Operators, Expressions, Precedence and Associativity of Operators, Type Conversion and Casting, Command Line Arguments.

Control Statements: Introduction, if Expression, Nested if Expressions, if–else Expressions, Switch Statement, Iteration Statements, while Expression, do–while Loop, for Loop, Nested for Loop, For–Each for Loop, Break Statement, Continue Statement, return statement.

UNIT-II:

Classes and Objects: Introduction, Class Declaration, Declaration of Class Objects, Assigning One Object to Another, Access Control for Class Members, Accessing Private Members of Class, Defining Methods, Overloaded Methods, Constructor Methods for Class, Overloaded Constructor Methods, Nested Classes, final Class and Methods, Passing Arguments by Value and by Reference, Keyword this, Recursive Methods, Nesting of Methods, Static Variables and Methods, Arrays, String Handling in Java.

.**UNIT-III:**

Inheritance: Introduction, Types of Inheritance, Access Control and Inheritance, Keyword Super, Constructor Method and Inheritance, Method Overriding, Abstract Classes.

Interfaces: Introduction, Declaration of Interface, Implementation of Interface, Multiple Interfaces, Nested Interfaces, Inheritance of Interfaces, default methods and static methods in interface.

Packages and Java Library: Introduction, Defining Package, Importing Packages and Classes into Programs, Path and Class Path, Access protection, Java.lang Package and its Classes, Wrapper Classes.

UNIT-IV:

Exception Handling: Introduction, Keywords try, catch, throws, throw, and finally Blocks, Multiple Catch Clauses, Unchecked Exceptions, Checked Exceptions, Nested try and catch Blocks, User-defined Exception.

Multithreading : The Java Thread Model, Creation of new threads, Thread Priorities, The Thread Class and the Runnable Interface, Creating Multiple Threads, Synchronization, Inter-thread Communication.

UNIT-V:

Applets: Introduction, Applet Life Cycle, Applet class, Simple Applet Display Methods.

Event Handling: Introduction, Event Delegation Model, Event Listeners, Event Classes, Handling Mouse Events, Handling Keyboard Events.

AWT: Introduction, AWT Classes, AWT Control Fundamentals: Labels, Using Buttons, Applying Check Boxes, Choice Controls, Using Lists, Managing Scroll Bars, Using a TextField, Using a TextArea, Understanding Layout Managers, Menu Bars and Menus.

Text Books:

1. The Complete Refernce Java, 9ed, Herbert Schildt, TMH

2. Programming in JAVA, Sachin Malhotra, Saurabh choudhary, Oxford.

3. JAVA for Beginners, 4e, Joyce Farrell, Ankit R. Bhavsar, Cengage Learning.

4. JAVA one step ahead, Anitha Seth, B.L.Juneja, Oxford

5. Introduction to Java rogramming, 7 th ed, Y Daniel Liang, Pearson

Reference Books:

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, Universities Press.

5. Murach's Java Programming, Joel Murach

CO-PO MAPPING

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO.1	**2**	**2**	**2**	**1**	**2**	**1**	**-**	**-**	**1**	**1**	**1**	**1**
CO.2	**2**	**3**	**2**	**1**	**1**	**1**	**-**	**1**	**1**	**1**	**1**	**1**
CO.3	**2**	**2**	**2**	**2**	**2**	**-**	**-**	**-**	**1**	**1**	**1**	**1**
CO.4	**2**	**3**	**3**	**3**	**2**	**1**	**-**	**1**	**1**	**1**	**1**	**2**
CO.5	**2**	**3**	**3**	**3**	**2**	**1**	**-**	**1**	**1**	**1**	**1**	**2**

ST. ANN'S COLLEGE OF ENGINEERING & TECHNOLOGY: CHIRALA

(AUTONOMOUS)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Year & Sem	II Year – II Semester				
Course Code	22UCS20	L	T	P	C
		3	0	0	3
Course Name	FORMAL LANGUAGES AND AUTOMATA THEORY				

Course Objectives:

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

Course Outcomes:

By the end of the course students can

- Employ finite state machines to solve problems in computing
- Summarize language classes & grammars relationship among them with the help of Chomsky hierarchy
- Illustrate the CFG grammar and normalizing the grammar.
- Design the PDA for acceptance
- Illustrate the TM and relation between CFG, PDA and TM.

UNIT I

Finite Automata: Need of Automata theory, Central Concepts of Automata Theory, Finite Automation, Acceptance of a String, DFA, Design of DFAs, NFA, Design of NFA, Equivalence of DFA and NFA, Conversion of NFA into DFA, Finite Automata with Є-

Transitions, Minimization of Finite Automata, Finite Automata with output-Mealy and Moore Machines, Applications and Limitation of Finite Automata.

UNIT II

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

UNIT III

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

UNIT IV

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

UNIT V

Turning Machine: Definition, Model, Language of a TM, Design of TMs, Types of TMs, Church's Thesis, Universal and Restricted TM, Decidable and Un-decidable Problems, Halting Problem of TMs,

Text Books:

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

Reference Books:

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

Hill,2014

e-Resources:

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

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO.1	3	3	3	2	-	2	-	-	-	-	-	2
CO.2	3	3	3	2	-	2	-	-	-	-	-	2
CO.3	3	3	3	2	-	2	-	-	-	-	-	2
CO.4	3	3	3	3	-	2	-	-	-	-	-	2
CO.5	3	3	3	3	-	2	-	-	-	-	-	2

ST. ANN'S COLLEGE OF ENGINEERING & TECHNOLOGY: CHIRALA

(AUTONOMOUS)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Year & Sem	II Year – II Semester				
Course Code	22UCS21	L	T	P	C
		0	0	3	1.5
Course Name	JAVA PROGRAMMING LAB				

Course Objectives:

The aim of this lab is to

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

Course Outcomes:

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

- Evaluate default value of all primitive data type, Operations, Expressions, Controlflow, Strings
- Determine Class, Objects, Methods, Inheritance, Exception, Runtime Polymorphism, User
- defined Exception handling mechanism

Illustrating simple inheritance, multi-level inheritance, Exception handling mechanism

- Construct Threads, Event Handling, implement packages, developing applets

Illustrating GUI programming with AWT and Swing

Exercise - 1 (Operations, Expressions, Control-flow, Strings)

a) Write a JAVA program to display the Fibonacci sequence

b) Write a JAVA program give example for command line arguments.

c) Write a JAVA program to sort an array of strings

Exercise - 2 (Class, Objects)

a) Write a JAVA program to implement class mechanism. Create a class, methods and invoke

them inside main method.

b) Write a JAVA program to implement constructor.

Exercise - 3(Methods)

a) Write a JAVA program to implement constructor overloading.

b) Write a JAVA program implement method overloading.

Exercise - 4 (Inheritance)

a) Write a JAVA program to implement Single Inheritance

b) Write a JAVA program to implement multi level Inheritance

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

Exercise - 5 (Inheritance - Continued)

a) Write a JAVA program give example for "super" keyword.

b) Write a JAVA program illustrating multiple inheritance using interfaces.

Exercise - 6 (Exception)

a) Write a JAVA program that describes exception handling mechanism

b) Write a JAVA program Illustrating Multiple catch clauses

Exercise – 7 (Runtime Polymorphism)

a) Write a JAVA program that implements Runtime polymorphism

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

Exercise – 8 (User defined Exception)

a) Write a JAVA program for creation of Illustrating throw

b) Write a JAVA program for creation of Java Built-in Exceptions

c) Write a JAVA program for creation of User Defined Exception

Exercise – 9 (Threads)

a) Write a JAVA program that creates threads by extending Thread class .First threaddisplay "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 JAVA program illustrating isAlive and join ()

Exercise - 10 (Threads continuity)

a) Write a JAVA program Producer Consumer Problem

b) Write a JAVA Program illustrating Daemon Threads.

Exercise – 11 (Packages)

a) Write a JAVA program illustrate class path

b) Write a case study on including in class path in your os environment of your package.

c) Write a JAVA program that import and use the defined your package in the previous Problem

Exercise - 12 (Applet)

a) Write a JAVA program to paint like paint brush in applet.

b) Write a JAVA program to create different shapes and fill colors using Applet.

Exercise - 13 (Event Handling)

a) Write a JAVA program that display the x and y position of the cursor movement using Mouse.

b) Write a JAVA program that identifies key-up key-down event user entering text in a Applet.

Exercise – 14 (AWT, SWING)

a) Write a JAVA program for creation of Illustrating Menu Bars and Menus in AWT.

b) Write a JAVA program to demonstrate working of GridLayout and BorderLayout in AWT

c) Write a JAVA program to implement the creation of simple Swing-based applet.

CO-PO MAPPING

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO.1	**2**	**2**	**2**	**1**	**1**	**-**	**-**	**1**	**-**	**1**	**-**	**1**
CO.2	**2**	**3**	**2**	**1**	**1**	**-**	**-**	**-**	**1**	**1**	**1**	**1**
CO.3	**2**	**2**	**2**	**2**	**1**	**-**	**-**	**-**	**1**	**1**	**1**	**1**
CO.4	**2**	**3**	**3**	**2**	**2**	**1**	**-**	**1**	**1**	**1**	**1**	**1**
CO.5	**2**	**3**	**3**	**3**	**2**	**1**	**-**	**1**	**1**	**1**	**1**	**1**

ST. ANN'S COLLEGE OF ENGINEERING & TECHNOLOGY: CHIRALA

(AUTONOMOUS)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Year & Sem	II Year – II Semester				
Course Code	22UCS22	L	T	P	C
		0	0	3	1.5
Course Name	OPERATING SYSTEMS AND LINUX LAB				

Course Objectives:

This Course will enable students to implement CPU scheduling algorithms, Disk scheduling algorithms, execute different types of Linux commands and Write shell scripts

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

- Implement various CPU scheduling algorithms and compare results
- Implement various disk scheduling algorithms and compare results
- Implement page replace algorithms
- Implement various memory management techniques.
- Execute basic Linux commands

List of Experiments:

UNIX Lab- Introduction to UNIX

1. Study of Unix/Linux general purpose utility commands
2. C program to emulate the UNIX ls –l command

3. C program that illustrates how to execute two commands concurrently with a command pipe. Ex: - ls –l | sort
4. Multiprogramming-Memory Management-Implementation of fork (), wait (), exec () and exit (), System calls

Operating Systems Lab

1. Simulate the Following CPU Scheduling Algorithms
   A) FCFS B) SJF C) Priority D) Round Robin
2. Multiprogramming-Memory Management- Implementation of fork (), wait (), exec () and exit ()
3. Simulate the Following
   a. Multiprogramming with A Fixed Number of Tasks (MFT)
   b. Multiprogramming with A Variable Number of Tasks (MVT)
4. Write a program to implement first fit, best fit and worst fit algorithm for memory management.
5. Simulate Bankers Algorithm for Dead Lock Avoidance
6. Simulate Bankers Algorithm for Dead Lock Prevention.
7. Simulate the Following Page Replacement Algorithms.
   a) FIFO b) LRU c) LFU
8. Simulate the Following File Allocation Strategies a) Sequenced b) Indexed c) Linked

Linux Lab

1. Write a Shell program to check whether given number is prime or not.
2. Write a shell script which will display Fibonacci series up to the given range.
3. Write a shell script to check whether the given number is Armstrong or not.
4. Write a shell script to accept student number, name, marks in 5 subjects.
5. Find total, average and grade using the following rules:

   Avg>=80 then grade A

   Avg<80&&Avg>=70 then grade B

   Avg<70&&Avg>=60 then grade C

   Avg<60&&Avg>=50 then grade D

   Avg<50&&Avg>=40 then grade E
6. Write a shell script to find minimum and maximum elements in the given list of elements.

7. Write a shell program to check whether the given string is palindrome or not.

8. Write an awk program to print sum, avg of students marks list.
9. Write a shell script to compute no. of characters and words in each line of given file.
10. Write a shell script to check whether the given input is a number or a string.

CO- PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO.1	**3**	**3**	**3**	**3**	**3**	**2**	**1**	**1**	**1**	**1**	**1**	**3**
CO.2	**3**	**3**	**2**	**3**	**2**	**2**	**2**	**1**	**1**	**1**	**1**	**3**
CO.3	**3**	**3**	**3**	**3**	**3**	**1**	**1**	**1**	**1**	**1**	**1**	**3**
CO.4	**3**	**3**	**2**	**3**	**2**	**3**	**3**	**2**	**1**	**1**	**2**	**3**
CO.5	**3**	**2**	**2**	**2**	**2**	**3**	**3**	**2**	**1**	**1**	**2**	**3**

ST. ANN'S COLLEGE OF ENGINEERING & TECHNOLOGY: CHIRALA

(AUTONOMOUS)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Year & Sem	II Year – II Semester				
Course Code	22UCS23	L	T	P	C
		0	0	3	1.5
Course Name	DATABASE MANAGEMENT SYSTEMS LAB				

Course Objectives:

This Course will enable students to

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

Course Outcomes:

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

- Utilize SQL to execute queries for creating database and performing data manipulation operations
- Examine and applying integrity constraints to build efficient databases
- Utilize SQL to execute queries for using Aggregate functions, SQL functions like string functions, conversion function
- Build PL/SQL programs including stored procedures, functions, cursors and triggers
- perform the search operation on table using indexing and non-indexing techniques.

List of Exercises:

1. Creation, altering and droping of tables and inserting rows into a table (use constraints while creating tables) examples using SELECT command.

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

Text Books/Suggested Reading:

1) Oracle: The Complete Reference by Oracle Press

2) Nilesh Shah, "Database Systems Using Oracle", PHI, 2007

3) Rick F Vander Lans, "Introduction to SQL", Fourth Edition, Pearson Education, 2007

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	1	2	3	3	3	2	-	-	-	-	-	2
CO2	1	2	3	3	3	2	-	-	-	-	-	2
CO3	1	2	3	3	3	2	-	-	-	-	-	2
CO4	1	2	3	3	3	2	-	-	-	-	-	2
CO5	1	2	3	3	3	2	-	-	-	-	-	2

ST. ANN'S COLLEGE OF ENGINEERING & TECHNOLOGY: CHIRALA

(AUTONOMOUS)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Year & Sem	II Year – II Semester				
Course Code	22UCS24	L	T	P	C
		1	0	2	2
Course Name	Skill Oriented Course WEB APPLICATION DEVELOPMENT USING HTML, CSS, XML				

Course Objectives:

The objective of this lab is to provide understanding about the core concepts of frontend programming for web application

Course Outcomes:

By the end of this lab the student is able to

- Analyze a web page and identify its elements and attributes.
- Demonstrate the important HTML tags for designing static pages and separate design from content using Cascading Style sheet
- Outline the concepts of Extensible markup language
- Implement MVC and responsive design to scale well across PC, tablet and Mobile Phone
- Create web pages using HTML and Cascading Style Sheets.

Perform experiments related to the following concepts:

A) HTML

1) Introduction to HTML
2) Browsers and HTML
3) Tags, Attribute and Elements
4) Headings, Paragraphs, and Formatting Text
5) Lists and Links
6) Iframes
7) Images and Tables
8) Forms

B) CSS

1) Introduction CSS
2) Applying CSS to HTML
3) Selectors, Properties and Values
4) CSS Colors and Backgrounds
5) CSS Box Model
6) CSS Margins, Padding, and Borders
7) Conflict Resolution

C) XML

1) Introduction to XML
2) Document type Definition (DTD)
3) XML schemas
4) XSLT
5) Parsers - DOM and SAX

CO-PO MAPPING

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO.1	**2**	**2**	**2**	**1**	**1**	-	-	-	-	-	-	-
CO.2	**2**	**2**	**2**	**1**	**1**	-	-	-	-	**1**	**1**	**1**
CO.3	**2**	**2**	**2**	**1**	**1**	-	-	-	-	**1**	**1**	**1**
CO.4	**2**	**2**	**2**	**1**	**1**	-	-	-	-	**1**	**1**	**1**
CO.5	**2**	**2**	**2**	**1**	**1**	-	-	-	-	**1**	**1**	**1**