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**

COURSE STRUCTURE FOR COMPUTER SCIENCE AND ENGINEERING

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**

COURSE STRUCTURE FOR COMPUTER SCIENCE AND ENGINEERING

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**

COURSE STRUCTURE FOR COMPUTER SCIENCE AND ENGINEERING

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

COURSE STRUCTURE FOR COMPUTER SCIENCE AND ENGINEERING

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	22UCS31	Computer Networks	3	0	0	30	70	100	3
2	Humanities and Social Sciences	22UCS32	Managerial Economics and Financial Accountancy	3	0	0	30	70	100	3
3	Professional Core courses	22UCS33	Data Warehousing and Data Mining	3	0	0	30	70	100	3
4	Open Elective Course/Job oriented	22UCS51	**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	22UCS34	**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	22UCS38	Data Warehousing and Data Mining Lab	0	0	3	30	70	100	1.5
7	Professional Core courses Lab	22UCS39	Full Stack Technologies Lab	0	0	3	30	70	100	1.5
8	**Skill advanced course/ soft skill course***	22UCS40	**Skill Oriented Course** Animation course: Animation Design	0	1	2	15	35	50	2
9	Mandatory course (AICTE suggested)	22UEN15	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)22UOC02**			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

COURSE STRUCTURE FOR COMPUTER SCIENCE AND ENGINEERING

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	22UCS41	Machine Learning	3	0	0	30	70	100	3
2	Professional Core courses	22UCS42	Advanced data structures and algorithms	3	1	0	30	70	100	4
3	Professional Core courses	22UCS43	Cryptography and Network Security	3	0	0	30	70	100	3
4	Professional Elective courses	22UCS46	**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	22UCS53	**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	22UCS48	Machine Learning using Python Lab	0	0	2	15	35	50	1
7	Professional Core courses Lab	22UCS49	MEAN Stack Development Lab	0	0	3	15	35	50	1.5
8	Professional Core courses Lab	22UCS50	ADSA Lab	0	0	2	15	35	50	1
9	**Skill advanced course/ soft skill course***	22UCS52	**Skill Oriented Course - IV** Android Lab	1	0	2	-	50	50	2
10	Mandatory course (AICTE suggested)	22UEN16	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 Pool, as per the opted track	**4**	**0**	**0**	**70**	**30**	**100**	**4**

Minor course through SWAYAM	-	-	-	-	-	-	**2**

$- Integrated Course

COURSE STRUCTURE FOR COMPUTER SCIENCE AND ENGINEERING

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

COURSE STRUCTURE FOR COMPUTER SCIENCE AND ENGINEERING

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-worldproblems 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 areasbounded by region.
CO5: Student will learn important tools of calculus in higherdimensions. 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:

Attheendofthemodule, thelearnerswillbeableto

- 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.
- Employsuitablestrategiesforskimmingandscanningtogetthegeneralideaofatextandlocate specificinformation.
- Recognizeparagraphstructureandbeabletomatchbeginnings/endings/headingswithparagraphs
- Formsentencesusingproper grammaticalstructuresandcorrectwordforms.

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 toprose andconversations. **Speaking:**Askingandansweringgeneralquestionsonfamiliartopicssuchashome,family, work studiesandinterests. Self-introductionandintroducingothers. **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, signpostsandtransitionsignals;mechanicsofwriting-punctuation,capitalletters.

Vocabulary: Synonyms and Antonyms,Affixes.

Grammar: Content words and function words, word forms.

Unit-2

Lesson-1:Nehru'slettertohisdaughterIndiraonherbirthdayfrom "**InfotechEnglish**",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; recognizingverbaltechniques thathelptolinktheideasina paragraphtogether. **Writing**: Summarizing - identifying main idea and rephrasing what is read; avoidingredundanciesandrepetitions. **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 speakingandwriting. **Speaking:**Discussing specifictopicsinpairsorsmallgroupsandreportingwhatisdiscussed. FunctionalEnglish: ComplainingandApologizing. **Reading**: Reading a text in detail by making basic inferences - recognizing and interpreting specificcontextclues;strategiestousetextcluesforcomprehension. Criticalreading. **Writing**: Letterwriting-types,formatandprinciplesofletterwriting. E-mailetiquette,WritingCV's. **Vocabulary**: Synonyms and Antonyms, Word Formation

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

Unit4

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

Listening: Making predictions while listening to conversations/ transactional dialogues without video(onlyaudio);listeningtoaudio-visualtexts. **Speaking:**RoleplaysforpracticeofconversationalEnglishinacademiccontexts(formalandinformal) - asking for and giving information/directions. Functional English: Permissions, Requesting,Inviting. **Reading**:Studyingtheuseofgraphicelementsintextstoconveyinformation, revealtrends/patterns/relationships,communicative processordisplaycomplicateddata. **Writing**: Information transfer; describe, compare, contrast, identify significance/trendsbasedoninformationprovidedinfigures/charts/graphs / tables, writingfor media. **Vocabulary**: Synonyms and Antonyms, Phrasal verbs.

Grammar:Tenses, correction of sentences.

Unit5

Lesson-1:StayHungry-Stayfoolishfrom"**InfotechEnglish**", MaruthiPublications

Listening: Identifyingkeyterms,understandingconceptsandinterpretingtheconceptsbothinspeakingandwriting, TEDX Videos. **Speaking:**Formaloralpresentationsontopicsfromacademiccontexts-withouttheuseofPPTslides. FunctionalEnglish:Suggesting/Opiniongiving. **Reading**:Readingforcomprehension.RAPStrategy IntensivereadingandExtensivereadingtechniques. **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.

Suggestedbooks:

1. **InfotechEnglish**",MaruthiPublications.
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**

CourseObjectives:

Theobjectives of ProgrammingforProblem SolvingUsingC are

- Tolearnaboutthe computersystems,computingenvironments,developingof acomputerprogramand StructureofaC Program
- Togainknowledgeoftheoperators, selection,control statementsandrepetition inC
- Tolearnaboutthedesignconceptsofarrays,strings
- Toassimilateaboutpointers,dynamicmemoryallocation
- ToassimilateaboutFile, I/Oandsignificanceoffunctions

CourseOutcomes:

Uponthecompletionofthecoursethestudentwill 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

IntroductiontoComputers: ComputerSystems-BlockDiagramofComputer,Hardware,Software,Algorithms, Flow Charts, Pseudocode **IntroductiontotheCLanguage:**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&MakingDecisions:**LogicalDataandOperators,TwoWaySelection,MultiwaySelection,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,UsingArrayinC,ArrayApplication,TwoDimensionalArrays,Multidimensional

Arrays, ProgrammingExamples.
Strings: StringConcepts,CString,StringInput/OutputFunctions,ArraysofStrings,StringManipulationFunctions, ProgrammingExamples.

UNIT-IV

Functions: Designing, Structured Programs, Function in C, User Defined Functions, Types offunctions, Standard Functions, Passing Array to Functions and Passing Pointers to Functions,Recursion,Scope-GlobalScope,LocalScope, FunctionScope, andStorageClasses.
Pointers: Introduction - Definition, Declaration, Initialization, Accessing, Benefits of Pointers,WhyPointers,Pointerstopointers,Arrays,andPointers,PointerArithmeticandArrays, MemoryAllocationFunction, ArrayofPointers, ProgrammingApplication.

UNIT-V

Structures and Union: Structure- Definition, Declaration, Accessing, Initialization, Arrays inStructures,ArrayofStructures,StructurePointers,structuresandfunctions,UnionsandProgrammingExamples. **Files:** Files, Streams, Types of Files- Text and Binary Files, Standard Library Input / OutputFunctions,FormattingInput/OutputFunctions,CharacterInput/OutputFunctions,ProgrammingExamples

TextBooks:

1) ProgrammingforProblemSolving,BehrouzA.Forouzan,RichardF.Gilberg,CENGAGE.
2) TheCProgramming Language, BrianW.Kernighan, DennisM.Ritchie,2e,Pearson.

ReferenceBooks:

1) ComputerFundamentalsandProgramming,SumithabhaDas,McGrawHill.
2) ProgramminginC,AshokN.Kamthane,AmitKamthane,Pearson.
3) ComputerFundamentalsandProgramminginC,PradipDey,ManasGhosh,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 improvethe communication skills and clarity of speech.
- To enhance effective communication skills.
- Enables learners to speak and communicate confidently.

CourseOutcomes:

Bythe 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.

<u>Unit1:</u>

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

<u>Unit2</u>

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

<u>Unit3:</u>

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

<u>Unit4:</u>

Just A Minute (JAM)

<u>Unit5:</u>

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.
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	-	-	-	-	-	-	-

Year & Sem	**I Year – I Semester**					
Course Code	**22UCS02**	**L**	**T**	**P**	**SS**	**C**
Course Name	**PROGRAMMING FOR PROBLEM SOLVING USING C LAB**	**0**	**0**	**3**	**0**	**1.5**

CourseObjectives**:**

- Applythe principles ofClanguageinproblem solving.
- Todesignflowcharts,algorithmsand knowinghowtodebugprograms.
- Todesign&developof Cprogramsusingarrays,stringspointers &functions.
- Toreviewthefileoperations.

CourseOutcomes:

Bythe end of the Lab, thestudent
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

Exercise1:

1. WriteaCprogramtoprintablockFusinghash(#),wheretheFhasaheightofsixcharacters andwidthoffiveandfour characters.
2. WriteaCprogramtocomputetheperimeterandareaofarectanglewithaheightof7inches andwidthof 5inches.

Exercise2:

1. WriteaCprogram tocalculate thedistancebetweenthe twopoints.
2. Write a C program that accepts 4 integers p, q, r, s from the user where r and s are positiveand p is even. If q is greater than r and s is greater than p and if the sum of r and s is greaterthan the sumofpandqprint"Correct values",otherwiseprint"Wrongvalues".

Exercise3:

1. Write a program in C which is a Menu-Driven Program to compute the area of the variousgeometricalshape.
2. WriteaCprogram to calculate thefactorial ofa givennumber.

Exercise4:

1.WriteaprograminCtodisplaythentermsofharmonicseriesandtheirsum.1+1/2+1/3 +1/4 + 1/5 ... 1/n terms.

2.WriteaC programto check whethera given numberis an Armstrongnumberornot.

Exercise5:

1.Writeaprogramtodisplayallprimenumberslessthann

2.Writeaprogram todisplaythefollowingoutputformat

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

Exercise6:

1.Write a program in C to separate odd and even integers in separate arrays.

2.Writeaprogram in C to sort elements of arrayin ascendingorder.

Exercise7:

1.WriteaprograminC formultiplication oftwo squareMatrices.

2.Writeaprogram inC to findtranspose of a given matrix

Exercise8:

1.Writeaprogram in Cto comparetwo stringswithout usingstringlibraryfunctions.

2.Writeaprogram in C to copyonestringto another string.

Exercise9:

1.Writeacprogram toread and displaythe detailsof an employeeusingstructure

2. Writeacprogram to demonstrate arrayof structures

Exercise10:

1.
WriteaprograminCtodemonstratetheuseof&(addressof)and*(valueataddress)operator.

2.WriteaC program to find sum of n elements entered byuser usingpointers

3.WriteaCProgram toStoreInformationUsing StructureswithDynamicallyMemoryAllocation

Exercise11:

1.Writeaprogramin C toswap elements usingcall byreference

2.Findfactorial ofgiven numberusingrecursion

3.Writeaprogram inC to get thelargest element ofanarrayusingthe function

Exercise12:

1.Writeaprogramin Cto appendmultiple linesat theend ofatext file.

2.Writeaprogram in C to copyafilein another name.

3.Writeaprogramin Cto removea filefrom 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

CourseObjectives:

Theobjectiveofthis labis to

- Explaintheinternalpartsofacomputer,peripherals,I/Oports,connectingcables
- DemonstratebasicDOScommands
- DescribeaboutAntivirustools
- DemonstrateOfficeToolssuchasWordprocessors,SpreadsheetsandPresentationtools

CourseOutcomes:

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

ListofExperiments:

UNIT-1

Block diagram of a computer, Identification of peripherals of a PC, Laptop, Server, Smartphones, prepare a report containing the block diagram along with the configuration of eachcomponentanditsfunctionality,Input/Outputdevices,I/OportsandInterfaces,MainMemory,CachememoryandSecondaryStorageDevices, DigitalStorageBasics,NetworkingComponents and Speeds.

Experiment -1: Identification of peripherals.

Experiment-2: Assembling,Dissemblingofacomputer.

UNIT-2

Software:Definition,Softwaretypes,ApplicationSoftware,SystemSoftware**.**

Experiment-1:DOS Commands.

UNIT-3

MSWord: Creating a Document, Formatting, Bullets and Numbering, Page Settings, HeaderandFooter,InsertWordArt,Clip Art,Tables.

Experiment -1: Demonstrate and practice on wordFormatting **(**Bold, Italic, Underline, Alignments, Fonts, Sizes,Headingsetc.,**)**

Experiment-2: DemonstrateandPracticeonPageSettings,Margins,Headerand Footer**.Experiment-3:**Demonstrate andPracticeonWordArt,ClipArt.

Experiment-4: Demonstrateand Practiceon TableCreation.

UNIT-4

MS Excel: Create work sheet and work book, Search for data with in aWeb, InsertRows,Columns, Hiding of Rows and Columns, Renaming of Worksheet, Adjust Row Height andColumnWidth, Create Tables.

Experiment-1: CreateaTable,Performsumandaverageofasheet.

Experiment-2: AdjustRowheightandColumnwidthofaTable aspertheRequirements

UNIT-5

MS PowerPoint: Create and Manage Presentation, Slide transition and Animation, WWW,WebBrowser, Virus, Antivirus, Creatingmails.

Experiment-1: CreateaPPTonaTopicofyour Choice.

Experiment-2: Createamailidwith yourRollNumber

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

Course Objectives:

- To Enable the student to understand the importance of Constitution

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

- 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, 44thEdition, 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, 10thEdition, 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. Analyzetypes of plastics, methods of fabrication, the differenttypesofcompositeplasticmaterialsandinterpretthemechanismof conduction inconductingpolymers.

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 differenttypesofcompositeplasticmaterialsandinterpretthemechanismof conduction inconductingpolymers.**

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

CourseObjectives:

Theobjectiveofthecourseisto

- Introducethefundamentalconcept ofdatastructuresandabstractdatatypes
- Emphasizetheimportanceofdatastructuresindevelopingandimplementingefficientalgorithms
- Describehowarrays,records,linkedstructures,stacks,queues,trees,andgraphsare representedinmemoryandused byalgorithms

CourseOutcomes:

Aftercompletingthiscourse,astudentwillbeableto:
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.

UNITI

DataStructures-Definition,ClassificationofDataStructures,OperationsonDataStructures,AbstractDataType (ADT),Preliminariesofalgorithms.TimeandSpacecomplexity.
Searching-Linearsearch, Binarysearch, Fibonaccisearch.
Sorting-Insertionsort,Selectionsort,Exchange(Bubblesort,quicksort),distribution(radixsort), merging (Mergesort) algorithms.

UNITII

Linked List: Introduction, Single linked list, Representation of Linked list in memory, Operationson Single Linked list-Insertion, Deletion, Search and Traversal, Reversing Single Linkedlist,ApplicationsonSingleLinkedlist-PolynomialExpressionRepresentation, Addition, SparseMatrix Representation using Linked List, Advantages and Disadvantages of Single Linked list,Double Linked list-Insertion,Deletion, CircularLinkedlist-Insertion,Deletion.

UNITIII

Stacks: Introduction to Stacks, Array Representation of Stacks, Operations on Stacks, linked listRepresentationofStacks,OperationsonLinkedStack, Applications-Reversing list,FactorialCalculation,Infixto PostfixConversion,EvaluatingPostfixExpressions.

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- CircularQueues,Deques,PriorityQueues.

UNITIV

Trees:BasicTerminologyinTrees,BinaryTrees-Properties,RepresentationofBinaryTreesusingArraysandLinkedlists.TreeTraversal–In order,Preorder,Post order.Heaps–Definition,Max Heap, Min Heap, Insertion and Deletion from Max Heap, Heap sort,BinarySearch Trees – BasicConcepts,BSTOperations:InsertionandDeletion.

UNITV

Graphs: Basic Concepts, Representations of Graphs-Adjacency Matrix and using Linked list,GraphTraversals(BFT&DFT),Applications MinimumSpanningTreeUsingPrims&KruskalsAlgorithm,Dijkstra'sshortest path,Transitiveclosure, Warshall'sAlgorithm.

TextBooks:

1) DataStructuresUsingC.2nd Edition. ReemaThareja,Oxford.
2) DataStructuresandalgorithmanalysis inC,2nd ed,MarkAllenWeiss.

ReferenceBooks:

1) FundamentalsofDataStructuresinC,2ndEdition,Horowitz,Sahni,UniversitiesPress.
2) Data Structures:A PseudoCode Approach,2/e,RichardF.Gilberg,BehrouzA. Forouzon,Cengage.
3) DataStructureswithC,SeymourLipschutzTMH

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

CourseObjectives:

TheObjectivesof PythonProgrammingare

- TolearnaboutPythonprogramminglanguagesyntax,semantics,andtheruntimeenvironment
- Tobefamiliarizedwithuniversalcomputerprogrammingconceptslikedatatypes,containers
- Tobefamiliarizedwithgeneralcomputerprogrammingconceptslikeconditionalexecution,loops&functions
- Tobefamiliarizedwithgeneralcodingtechniquesandobject-orientedprogramming

CourseOutcomes:

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

UNITI

Introduction: Introduction to Python, Program Development Cycle, Input, Processing, andOutput,Displaying Output with the Print Function, Comments, Variables, Reading Input from the Keyboard,PerformingCalculations, Operators. Typeconversions, Expressions, MoreaboutData Output.

Data Types, and Expression: Strings Assignment, and Comment, Numeric Data Types and CharacterSets,Usingfunctions andModules.

DecisionStructuresandBooleanLogic:if,if-else,if-elif-elseStatements,NestedDecisionStructures,ComparingStrings,LogicalOperators,BooleanVariables.

RepetitionStructures: Introduction,whileloop,forloop,CalculatingaRunningTotal,InputValidationLoops, NestedLoops

UNITII

Control Statement: Definite iteration for Loop Formatting Text for output, Selection if and if elseStatementConditional Iteration theWhileLoop

Strings and Text Files: Accessing Character and Substring in Strings, Data Encryption, Strings andNumberSystems, StringMethods Text Files.

UNITIII

ListandDictionaries: Lists,DefiningSimpleFunctions,Dictionaries

Design with Function: Functions as Abstraction Mechanisms, Problem Solving with Top-DownDesign, Design with Recursive Functions, Case Study Gathering Information from a File System,ManagingaProgram’s Namespace, HigherOrderFunction.

Modules: Modules,StandardModules,Packages.

UNITIV

File Operations: Reading config files in python, Writing log files in python, Understanding readfunctions, read (), readline () and readlines(), Understanding write functions, write() and writelines(),Manipulatingfilepointerusingseek, Programmingusingfileoperations

Object Oriented Programming: Concept of class, object and instances, Constructor, class attributesanddestructors,Realtimeuseofclassinliveprojects,Inheritance,overlappingandoverloadingoperators, Adding and retrieving dynamic attributes of classes, Programming using Oops support**DesignwithClasses:**ObjectsandClasses,DatamodelingExamples,CaseStudyAnATM,StructuringClasses withInheritanceand Polymorphism

UNITV

Errors and Exceptions: Syntax Errors, Exceptions, Handling Exceptions, Raising Exceptions, User-definedExceptions, DefiningClean-upActions, Redefined Clean-upActions.

Graphical User Interfaces: The Behavior of Terminal Based Programs and GUI -Based, Programs,CodingSimple GUI-Based Programs, OtherUseful GUIResources.

Programming: IntroductiontoProgrammingConceptswithScratch.

TextBooks

1) FundamentalsofPythonFirstPrograms,Kenneth.A.Lambert,Cengage.
2) PythonProgramming: AModernApproach,VamsiKurama,Pearson.

ReferenceBooks:

1) IntroductiontoPython Programming,Gowrishankar.S,VeenaA,CRCPress.
2) IntroductiontoProgrammingUsingPython,Y. DanielLiang,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 STRUCTURESUSING C LABORATORY**	**0**	**0**	**3**	**0**	**1.5**

CourseObjectives:

Theobjectiveofthislab isto

- Demonstratethedifferentdatastructuresimplementation.

CourseOutcomes:

Bythe endofthislabthestudentisable 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.

ListofExperiments:

Exercise-1(Searching)

a) WriteCprogramthatusebothrecursiveandnon-recursivefunctionstoperformLinearsearchforakeyvaluein agivenlist.
b) WriteCprogramthatuse bothrecursiveandnon-recursivefunctionstoperformBinarysearchforakeyvaluein agivenlist.

Exercise-2(Sorting-I)

a) WriteCprogramthatimplementBubblesort,tosortagivenlistofintegersinascendingorder
b) WriteCprogramthatimplementQuicksort,tosortagivenlistofintegersinascendingorder
c) WriteCprogramthatimplementInsertionsort,tosortagivenlistofintegersinascendingorder

Exercise-3(Sorting-II)

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

Exercise-4(SinglyLinkedList)

a) WriteaCprogramthat uses functionsto createasinglylinkedlist

b) WriteaCprogramthatusesfunctionstoperforminsertionoperationonasinglylinkedlist

c) WriteaCprogramthatuses functionstoperformdeletionoperationonasinglylinkedlist

d) WriteaCprogramtoreverseelementsofasinglelinkedlist.

Exercise-5(Stack)

a) WriteCprogramthatimplementstack(itsoperations)usingarrays

b) WriteCprogramthatimplementstack(itsoperations)usingLinkedlist

c) WriteaCprogramthatusesStackoperationstoevaluatepostfix expression

Exercise-6(Queue)

a) WriteCprogramthatimplementQueue(itsoperations) usingarrays.

b) WriteCprogramthatimplementQueue(itsoperations)usinglinkedlists

Exercise-7(BinaryTree)

d)WritearecursiveCprogramfortraversingabinarytreeinpreorder,in orderandpost order.

Exercise-8(BinarySearchTree)

a) WriteaC programtoCreatea BST

b) WriteaCprogramtoinsertanodeintoaBST.

c) WriteaC programto deleteanodefromaBST.

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

CourseObjectives:

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**

TheObjectivesof PythonProgrammingare

- TolearnaboutPythonprogramminglanguagesyntax,semantics,andtheruntimeenvironment
- Tobefamiliarizedwithuniversalcomputerprogrammingconceptslikedatatypes, containers
- Tobefamiliarizedwithgeneralcomputerprogrammingconceptslikeconditional execution, loops &functions
- Tobefamiliarizedwithgeneral codingtechniques andobject-orientedprogramming

CourseOutcomes:

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

ListofExperiments:

1) Write a program that asks the user for a weight in kilograms and converts it to pounds.Thereare2.2 pounds in akilogram.
2) Write a program that asks the user to enter three numbers (use three separate inputstatements). Create variables called totaland average that hold the sum and average ofthethreenumbers and print outthe values oftotal andaverage.
3) Writeaprogram thatuses aforloop to print the numbers 8, 11,14, 17, 20, .. . ,83,86, 89.
4) Writeaprogramthataskstheuserfortheirnameandhowmanytimestoprintit.Theprogramshould print outtheuser'snamethespecified numberof times.
5) Useaforloop to print a triangle like theonebelow. Allowtheuser to specifyhowhighthe triangle should be.

```
*
**
***
****
```

6) Generatearandomnumberbetween1and10.Asktheusertoguessthenumberandprintamessagebased onwhethertheyget it rightor not.
7) Writea programthatasksthe user fortwonumbers andprintsClose ifthe numbersarewithin.001 ofeach otherand not close otherwise.

8) Writeaprogramthataskstheusertoenterawordandprintsoutwhetherthatword containsanyvowels.

9) Write aprogramthataskstheusertoentertwostringsofthesame length.Theprogram should then check to see if the strings are of the same length. If they are not theprogram should print an appropriate message and exit. If they are of the same length, theprogram should alternate the characters of the two strings. For example, if the user entersabcde and ABCDE theprogramshould print out AaBbCcDdEe.

10) Write a program that asks the user for a large integer and inserts commas into itaccordingtothestandardAmericanconventionforcommasinlargenumbers.Forinstance,if theuser enters 1000000, theoutput should be 1,000,000.

11) Inalgebraicexpressions,thesymbolformultiplicationisoftenleftout,asin3x+4yor 3(x+5). Computers prefer those expressions to include the multiplication symbol, like3*x+4*y or 3*(x+5). Write a program that asks the user for an algebraic expression andtheninserts multiplication symbols whereappropriate.

11)Writeaprogramthatgeneratesalistof20randomnumbersbetween1and100.

(a) Printthelist.

(b) Printtheaverageof theelements inthe list.

(c) Printthelargest and smallest values in thelist.

(d) Printthesecondlargest andsecond smallestentries inthelist

(e) Print how manyevennumbers arein thelist.

12) Write a program that asks the user for an integer and creates a list that consists of thefactorsofthat integer.

13)Write a program that generates 100 random integers that are either 0 or 1. Then findthe longest run of zeros, the largest number of zeros in a row.For instance, the longestrunof 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 appearsatmost once. For instance,the list [1,1,2,3,4,3,0,0]wouldbecome [1,2,3,4,0].

15)Write a program that asks the user to enter a length in feet. The program should thengive the userthe optiontoconvertfromfeetintoinches,yards, miles,millimeters,centimeters, meters, or kilometers. Say if the user enters a 1, then the program converts toinches, if they enter a 2, then the program converts to yards, etc. While this can be donewith if statements, it is much shorter with lists and it is also easier to add new conversionsif you uselists.

16)Write a function called sum digits that is given an integer num and returns the sum ofthedigits of num.

17)Write a function called first_diff that is given two strings and returns the first locationinwhich thestrings differ.Ifthestrings areidentical, itshould return-1.

18)Write a function called number_of_factors that takes an integer and returns howmanyfactors the numberhas.

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

20)Write a function called root that is given a number x and an integer n and returnsx1/n.In thefunction definition, set thedefault value ofn to 2.

21)Write a function called primes that is given a number n and returns a list of the first nprimes.Let thedefault value ofn be100.

Write a function called merge that takes two already sorted lists of possibly

differentlengths,and merges theminto asingle sorted list.
(a)Do thisusingthesortmethod. (b) Dothis without usingthe sort method.
22)Write a program that asks the user for a word and finds all the smaller words that canbe made from the letters of that word. The number of occurrences of a letter in a smallerwordcan’t exceed the numberof occurrencesof theletterin theuser’sword.
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 bysemicolons.
24)Write a program that reads a list of temperatures from a file called temps.txt, convertsthosetemperatures toFahrenheit, andwrites theresults toa filecalled ftemps.txt.
25)Write a class called Product. The class should have fields called name, amount, andprice, holding the product’s name, the number of items of that product in stock, andtheregular price of the product. There should be a method get_price that receives the numberof items to be bought and returns the cost of buying that many items, where the regularprice is charged for orders of less than 10 items, a 10% discount is applied for orders ofbetween 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 tobeboughtand decreasesamount bythat much.
26)Write a class called Time whose only field is a time in seconds. It should have amethod called convert_to_minutes that returns a string of minutes and seconds formattedas in the following example: if seconds is 230, the method should return '5:50'. It shouldalso have a method called convert_to_hours that returns a string of hours, minutes, andsecondsformattedanalogouslyto theprevious method.
27)Write a class called Converter. The user will pass a length and a unit when declaringan object from the class—for example, c = Converter(9,'inches'). The possible units areinches, feet, yards, miles, kilometers, meters, centimeters, and millimeters. For each ofthese units there should be a method that returns the length converted into those units. Forexample, using the Converter object created above, the user could call c.feet() and shouldget0.75as the result.
28)WriteaPythonclasstoimplement pow (x,n).
29)WriteaPython classto reverse astringword byword.
30)Write a program that opens a file dialog that allows you to select a text file. Theprogramthen displays thecontents of thefileinatextbox.
31)WriteaprogramtodemonstrateTry/except/else.
32)Writeaprogramto demonstrate try/finallyand 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					

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

(AUTONOMOUS)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

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 OrdinaryDifferential Equations

Numerical Integration - Trapezoidal rule– Simpson's 1/3rd and 3/8th 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, 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. **Erwin Kreyszig,** Advanced Engineering Mathematics, 10th 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	-	-	-

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

(AUTONOMOUS)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

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

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

(AUTONOMOUS)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

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

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

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

(AUTONOMOUS)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

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

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

(AUTONOMOUS)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

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	-	-	-	-	-	-	-	-	-	-

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

(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

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

(AUTONOMOUS)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

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	-	-	-	-	-	-	-	-	-

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

(AUTONOMOUS)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

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

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

(AUTONOMOUS)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

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**				

CourseObjectives:

1. TheobjectiveofthislabistoacquireprogrammingskillsinPythonpackageNumPyand performmathematicalandstatisticaloperations.
2. Theobjectiveofthislabistounderstandthe fundamentalsofthePandaslibraryinPython and howitisusedtohandledataandalsodevelopbasicskillsin data analysisand visualization

CourseOutcomes:

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.

<u>NUMPY</u>

Performthefollowing:

1) NumPyInstallationusingdifferentscientificpythondistributions (Anaconda, Python(x,y),WinPython,Pyzo)

2) NumPyBasics(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) ArrayManipulation(np.append,np.insert,np.resize,np.delete,np.concatenate, np.vstack,np.hstack)

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

6) NumPydatatypes

7) NumPyndarray

8) NumPyStringOperations

9) NumPyFinancial functions

10)NumPyFunctionalProgramming

PANDAS

Performthefollowing:

1)PandasInstallation

2) CreatingDataFrames

Exercises:

A) PandasDataSeries:

1) Writea Pandasprogramtocreateanddisplayaone-dimensionalarray-likeobject containinganarrayofdatausingPandasmodule.

2) Writea PandasprogramtoconvertaPandamoduleSeriestoPythonlistandit'stype.

3) WriteaPandasprogramtoadd,subtract,multipleanddividetwoPandasSeries.

4) Writea PandasprogramtoconvertaNumPyarraytoaPandasseries.

SampleSeries: NumPyarray:

[10 20 30 40 50] ConvertedPandas series:

0 10

1 20

2 30

```
3 40
4 50
dtype:int64
```

B)PandasDataFrames:

Consider SamplePythondictionarydataandlistlabels:

```
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) Writea PandasprogramtocreateanddisplayaDataFramefromaspecified dictionarydatawhichhas theindexlabels.

2) Writea Pandasprogramtochangethename'James'to'Suresh'innamecolumnofthe Data Frame.

3) WriteaPandasprogramtoinsertanewcolumninexistingData Frame.

4) WriteaPandasprogramtogetlistfromData Framecolumnheaders.

5) WriteaPandasprogramtogetlistfromData Framecolumnheaders. C) PandasIndex:

1) Writea Pandasprogramtodisplaythedefaultindexand setacolumnasanIndexinagivendata frame.

2) Writea Pandasprogramtocreateanindexlabelsbyusing64-bitintegers,using floating-pointnumbersinagivendata frame.

D)PandasStringand Regular Expressions:

1) Writea Pandasprogramtoconvertallthestringvaluestoupper, lowercasesina givenpandasseries.Alsofindthelengthofthestringvalues.

2) WriteaPandasprogramtoremovewhitespaces,leftsidedwhitespacesandrightsided whitespacesofthestringvalues ofagivenpandas series.

3) Writea Pandasprogramtocountofoccurrenceofaspecifiedsubstringina Data Framecolumn.

4) Writea Pandasprogramtoswapthecases ofaspecifiedcharactercolumninagiven Data Frame.

E)PandasJoiningandmergingData Frame:

1) Writea Pandasprogramtojointhetwogivendata framesalongrowsandassignall data.

2) Writea Pandasprogramtoappendalistofdictioneriesorseriestoaexisting DataFrameanddisplaythecombineddata.

3) Writea Pandasprogramtojointhetwodataframeswithmatchingrecordsfromboth sideswhereavailable.

F)PandasTimeSeries:

1) WriteaPandas programtocreate

a)DatetimeobjectforJan152012.

b)Specificdateandtimeof9:20pm.

c)Localdateandtime.

d)Adatewithouttime.

e)Currentdate.

f)Timefromadatetime.

g)Currentlocaltime.

2) Writea Pandas programtocreateadatefromagivenyear,month,dayandanother datefromagivenstringformats.

3) Writea Pandas programtocreateatime-series withtwoindexlabels andrandom values.Alsoprintthetypeoftheindex.

	school	class	name	Date_of_birth	Age	Height	Weigh	address
S1	s001	V	AlbertoFranco	15/05/2002	12	173	35	street1
S2	s002	V	GinoMcneill	17/05/2002	12	192	32	street2
S3	s003	VI	RyanParkes	16/02/1999	13	186	33	street3
S4	s001	VI	EeshaHinton	25/09/1998	13	167	30	street1
S5	s002	V	GinoMcneill	11/05/2002	14	151	31	street2
S6	s004	VI	DavidParkes	15/09/1997	12	159	32	street4

G)PandasGroupingAggregate: Considerdataset:

1) Writea Pandasprogramtosplitthefollowingdataframeintogroupsbasedonschool code. AlsocheckthetypeofGroupByobject.

2) Writea Pandas programtosplitthefollowingdataframebyschoolcodeandget mean,min,andmaxvalueofageforeachschool.

H)PandasStyling:

1) Createadataframeoftenrows,fourcolumnswithrandomvalues.Writea Pandas programtohighlightthenegativenumbers redandpositivenumbersblack.

2) Createadataframeoftenrows,fourcolumnswithrandomvalues.Writea Pandas programtohighlightthemaximumvalueineachcolumn.

3) Createadataframeoftenrows,fourcolumnswithrandomvalues.Writea Pandas programtohighlightdataframe'sspecificcolumns.

I)Excel:

1) Writea Pandasprogramtoimportexceldataintoa Pandasdataframe.

2) Writea Pandasprogramtofindthesum,mean,max,minvalueofacolumnoffile.

J)Plotting:

1) Writea Pandasprogramtocreateahorizontalstackedbar plotofopening,closing stockprices ofanystockdatasetbetweentwospecificdates.

2) Writea Pandasprogramtocreateahistogramsplotofopening,closing,high,low stockpricesofstockdatasetbetweentwospecificdates.

3) Writea Pandasprogramtocreateastackedhistogramsplotofopening,closing,high, lowstockpricesofstockdatasetbetweentwospecificdateswithmorebins.

K) PandasSQLQuery:

1) Writea Pandasprogramtodisplayalltherecordsofastudentfile.

2) WriteaPandas

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**				

CourseObjectives:

1. Thecourseaimsatimpartingbasicprinciplesofthoughtprocess,reasoningandinferencing.Sustainabilityisthecoreof IndianTraditionalKnowledgeSystemsconnectingsocietyandnature.
2. HolisticlifestyleofYogic-scienceandwisdomcapsulesinSanskritliteraturearealsoimportantinmodernsocietywithrapidtechnologicaladvancementsandsocietaldisruptions.
3. ThecoursefocusesonintroductiontoIndianKnowledgeSystem,Indianperspectiveofmodernscientificworld-viewandbasic principlesofYogaandholistic healthcaresystem

CourseOutcomes:

Uponsuccessfulcompletionofthe course,the studentwill be able to:

1. UnderstandthesignificanceofIndianTraditionalKnowledge
2. ClassifytheIndianTraditionalKnowledge
3. CompareModernSciencewithIndianTraditionalKnowledgesystem.
4. AnalyzetheroleofGovernmentinprotectingtheTraditionalKnowledge
5. UnderstandtheimpactofPhilosophicaltraditiononIndian KnowledgeSystem.

UnitI

IntroductiontoTraditionalKnowledge:DefineTraditionalKnowledge-NatureandCharacteristics-ScopeandImportance-kindsofTraditionalKnowledge-ThehistoricalimpactofsocialchangeonTraditionalKnowledge Systems-Value ofTraditionalknowledgeinglobaleconomy.

UnitII

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),4upanga (Dharmashastra,Meemamsa,purana &TharkaShastra).

Unit III

ModernScienceandIndianKnowledgeSystem-IndigenousKnowledge,Characteristics-YogaandHolistic Healthcare-cases studies.

UnitIV

Protection of Traditional Knowledge: The need for protecting traditional knowledge -Significance ofTraditionalknowledge Protection-Role ofgovernmenttoharnessTraditionalKnowledge.

UnitV

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

ReferenceBooks:

1. TraditionalKnowledgeSysteminIndia,byAmitJha,2009.
2. TraditionalKnowledgeSystemandTechnologyinIndiabyBasantaKumarMohantaandVipinKumarSingh,PratibhaPrakashan2012.
3. Sivaramakrishnan(Ed.),CulturalHeritageofIndia-coursematerial,BharatiyaVidya
4. SwamiJitatmanand,HolisticScienceandVedant,BharatiyaVidyaBhavan
5. YogaSutraofPatanjali, RamakrishnaMission,Kolkata.
6. PramodChandra,IndiaArts,HowardUniv.Press,1983.
7. KrishnaChaitanya,ArtsofIndia, AbhinavPublications, 1987.

WebResources:

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												

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

(AUTONOMOUS)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

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

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 throughCorrelation and Regression tools.

CO4: Estimate the confidence intervals for the mean of a populationand TestaHypothesis concerning means.

CO5: Design the components of a classical Hypothesis tests andDerive 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 HypothesisIntroduction – 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.

1. [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]
2. **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.

3. 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**				

CourseObjectives:

Theobjectivesofthis courseisto

- Introducetotheinternaloperationofmodernoperating systems
- Define,explain,processesandthreads,mutualexclusion,CPUscheduling,deadlock , memorymanagement,andfilesystems
- UnderstandFileSystemsinOperatingSystemlikeUNIX/LinuxandWindows
- UnderstandInputOutputManagementanduseofDeviceDriverandSecondary Storage (Disk)Mechanism
- AnalyzeSecurityandProtectionMechanismin OperatingSystem

CourseOutcomes:

Afterlearning, thecoursethestudentsshouldbeableto:

- DescribevariousgenerationsofOperatingSystemandfunctionsofOperatingSystem
- Describetheconceptofprogram,processandthreadandanalyzevariousCPU SchedulingAlgorithmsandcomparetheirperformance
- SolveInterProcessCommunicationproblemsusingMathematicalEquationsby variousmethods
- ComparevariousMemoryManagementSchemesespeciallypagingandSegmentation inOperatingSystemandapplyvarious PageReplacementTechniques
- OutlineFileSystemsinOperatingSystemlikeUNIX/LinuxandWindows

UNITI

Operating Systems Overview: Operating system functions, Operating system structure, Operatingsystemsoperations,Computingenvironments,Open-SourceOperatingSystems. System Structures: Operating System Services, User and Operating-System Interface, systemscalls,Types ofSystem Calls,systemprograms,operatingsystem structure, operating systemdebugging,SystemBoot.

UNITII

ProcessConcept:Process scheduling,Operationsonprocesses,Inter-process communication, Communicationinclientserversystems.

MultithreadedProgramming:Multithreadingmodels,Threadlibraries,Threadingissues.

ProcessScheduling:Basicconcepts,Scheduling criteria,Schedulingalgorithms,Multiple processor scheduling,Threadscheduling.

UNITIII

Inter-processCommunication:Raceconditions,CriticalRegions,Mutualexclusionwithbusy waiting, Sleep and wakeup, Semaphores, Mutexes, Monitors, Messagepassing, Barriers, ClassicalIPC Problems - Diningphilosophers problem,Readers andwritersproblem.

Deadlocks:Resources,Conditionsforresourcedeadlocks,Ostrichalgorithm,Deadlock detectionandrecovery,Deadlockavoidance,Deadlockprevention.

UNITIV

Memory-ManagementStrategies:Introduction,Swapping,Contiguousmemoryallocation, Paging,Segmentation.

Virtual Memory Management: Introduction, Demandpaging,Copyon-write,Page replacement,Frameallocation, Thrashing,Memory-mappedfiles,Kernelmemoryallocation.

UNITV

FileSystems:Files, Directories,Filesystemimplementation,managementandoptimization. Secondary-StorageStructure:Overviewofdiskstructure,andattachment,Diskscheduling, RAIDstructure,Stablestorageimplementation.

SystemProtection:Goalsofprotection,Principlesanddomainofprotection,Accessmatrix, Accesscontrol,Revocationofaccess rights.

Case Studies: Linux, Microsoft Windows.

TextBooks:

1) SilberschatzA,GalvinPB,andGagneG,OperatingSystemConcepts,9thedition, Wiley,2013.
2) TanenbaumAS,ModernOperatingSystems,3rdedition,PearsonEducation,2008.

(forInterprocess CommunicationandFilesystems.)

ReferenceBooks:

1) DhamdhereDM,OperatingSystemsAConceptBasedApproach,3rdedition,Tata McGraw-Hill,2012.
2) StallingsW,OperatingSystems-InternalsandDesignPrinciples,6thedition,Pearson Education,2009
3) NuttG, OperatingSystems,3rdedition,PearsonEducation,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**

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

(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**				

CourseObjectives:

- TolearnfundamentalsofRegularandContextFreeGrammarsandLanguages TounderstandtherelationbetweenRegular LanguageandFiniteAutomatamachines
- TolearnhowtodesignAutomata'sandmachinesasAcceptors,VerifiersandTranslators
- TounderstandtherelationbetweenContextsfree Languages,PDAand TM
- To learnhow todesignPDAasacceptorandTMasCalculators

CourseOutcomes:

Bytheendofthecoursestudentscan

- Employfinitestatemachinestosolveproblems incomputing
- Summarizelanguageclasses&grammarsrelationship among themwiththehelpofChomskyhierarchy
- Illustrate the CFG grammar and normalizing the grammar.
- Design the PDA for acceptance
- Illustrate the TM and relation between CFG,PDA and TM.

UNITI

FiniteAutomata:NeedofAutomatatheory,CentralConceptsofAutomataTheory,Finite Automation, Acceptance of a String, DFA, Design ofDFAs, 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 withoutput-Mealyand Moore Machines,Applicationsand

LimitationofFiniteAutomata.

UNITII

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

UNITIII

Formal Languages,ContextFreeGrammar,LeftmostandRightmostDerivations,ParseTrees,SimplificationofContextFreeGrammars-EliminationofUselessSymbols,Є-ProductionsandUnit Productions,NormalForms-ChomskyNormalForm and Greibach Normal Form, Pumping Lemma, Closure Properties, Applications ofContextFreeGrammars.

UNITIV

PushdownAutomata,Definition,Model,GraphicalNotation,InstantaneousDescription,Language Acceptance of Pushdown Automata, Design of Pushdown Automata, Deterministicand Non – Deterministic Pushdown Automata, Equivalence of Pushdown Automata andContextFreeGrammars,Conversion,ApplicationofPushdown Automata.

UNITV

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

TextBooks:

1) Introduction to Automata Theory, Languages and Computation, J. E. Hopcroft, R. Motwani and J. D. Ullman, 3rd Edition, Pearson, 2008
2) TheoryofComputerScience-Automata, Languagesand Computation, K. L. P.MishraandN.Chandrasekharan,3rdEdition,PHI,2007

ReferenceBooks:

1) ElementsofTheoryofComputation,LewisH.P.&Papadimition C.H.,Pearson/PHI
2) TheoryofComputation,V.Kulkarni,OxfordUniversityPress,2013
3) TheoryofAutomata,LanguagesandComputation,RajendraKumar,McGrawHill,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**				

CourseObjectives:

ThisCoursewillenablestudentstoimplementCPUschedulingalgorithms,Diskschedulingalgorithms, executedifferenttypesofLinuxcommandsandWriteshellscripts

CourseOutcomes (COs):At theendof thecourse,studentwillbe ableto

- ImplementvariousCPUschedulingalgorithmsandcompareresults
- Implementvariousdiskschedulingalgorithms andcompareresults
- Implementpagereplacealgorithms
- Implementvariousmemorymanagementtechniques.
- ExecutebasicLinuxcommands

ListofExperiments:

UNIXLab-IntroductiontoUNIX

1. StudyofUnix/Linuxgeneralpurposeutilitycommands
2. CprogramtoemulatetheUNIXls–lcommand
3. Cprogramthatillustrateshowtoexecutetwocommandsconcurrentlywithacommandpipe. Ex:-ls –l1sort
4. Multiprogramming-Memory Management-Implementation offork(), wait(), exec() andexit(), Systemcalls

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**

Year & Sem	III Year – I Semester				
Course Code	22UCS31	L	T	P	C
Course Name	COMPUTER NETWORKS				

OBJECTIVES:

•To understand theprotocollayeringand physicallevelcommunication.

•To analyse theperformanceofanetwork.

•To understand thevarious componentsrequiredtobuild different networks.

•To learn thefunctionsof network layerand thevariousroutingprotocols.

•To familiarize thefunctions and protocolsof theTransportlayer.

UNITI

Networks–NetworkTypes–ProtocolLayering–TCP/IPProtocolsuite–OSIModel–Physical Layer: Performance– Transmissionmedia–Switching– Circuit-switched Networks– Packet Switching.

UNITII

Introduction–Link-LayerAddressing–DLCServices–Data-LinkLayerProtocols–HDLC–PPP-MediaAccessControl-WiredLANs:Ethernet-WirelessLANs–Introduction–IEEE802.11, Bluetooth –ConnectingDevices.

UNITIII

NetworkLayerServices–Packetswitching–Performance–IPV4Addresses–Forwardingof IPPackets-NetworkLayerProtocols:IP,ICMPv4–UnicastRouting Algorithms–Protocols–MulticastingBasics–IPV6Addressing–IPV6Protocol.

UNIT IV

Introduction–TransportLayerProtocols–Services–PortNumbers–UserDatagramProtocol–Transmission ControlProtocol–SCTP.

UNITV

WWWand HTTP –FTP –Email–Telnet–SSH–DNS–SNMP.

OUTCOMES:

On Completionof thecourse,thestudentsshouldbeable to:

- Understandthebasiclayers anditsfunctions incomputer networks.
- Evaluatetheperformanceof anetwork.
- Understandthebasicsof howdataflowsfromonenodetoanother.
- Analyze anddesignroutingalgorithms.
- Designprotocols for various functionsin the network.
- Understandtheworkingofvariousapplicationlayerprotocols.

TEXTBOOK:

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

REFERENCES

1. LarryL. Peterson,BruceS.Davie,Computer Networks:A Systems Approach,Fifth Edition,MorganKaufmannPublishers Inc.,2012.

2. WilliamStallings,Data andComputerCommunications, Tenth Edition,PearsonEducation,2013.

3. NaderF.Mir, ComputerandCommunicationNetworks, SecondEdition,PrenticeHall, 2014.

4. Ying-DarLin, Ren-HungHwangandFred Baker,Computer Networks:AnOpen-SourceApproach, McGrawHillPublisher,2011.

5. JamesF.Kurose,Keith W. Ross,ComputerNetworking, A Top-Down ApproachFeaturingtheInternet, SixthEdition,PearsonEducation,2013

CO-PO MAPPING:

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

Year & Sem	III Year – I Semester				
Course Code	22UCS32	L	T	P	C
Course Name	MANAGERIAL ECONOMICS AND FINANCIAL ANALYSIS				

Course Objectives:

- To inculcate the basic knowledge of micro economics and financial accounting

- To make the students learn how demand is estimated for different products, input-output relationship for optimizing production and cost

- To Know the Various types of market structure and pricing methods and strategy

- To give an overview on investment appraisal methods to promote the students to learn how to plan long term investment decisions.

- To provide fundamental skills on accounting and to explain the process of preparing financial statements.

Course Outcomes:

- Define the concepts related to Managerial Economics, financial accounting and management.

- Understand the fundament also of Economics viz., Demand, Production, cost, revenue and markets

- Apply the Concept of Production cost and revenues for effective Business decision

- Analyze how to invest their capital and maximize returns

- Evaluate the capital budgeting techniques

- Develop the accounting statements and evaluate the financial performance of business entity.

UNIT-I

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

Management.(relationship with others)

UNIT-II

Production and Cost Analysis: Introduction – Nature, meaning, significance, functions. Production

Function– Least-cost combination–Iso quant's and Iso costs, - Laws of Returns - Internal and External Economies of scale.

Cost & Break-Even Analysis - Cost concepts and Cost behavior- Break-Even Analysis (BEA) - Determination of Break-Managerial significance and limitations of Break- Even Analysis. Even Point (Simple Problems).

UNIT-III

Business Organizations and Markets: Introduction–Nature, meaning, significance, Functions

and advantages. Forms of Business Organizations- Sole Proprietary - Partnership - Joint Stock Companies -Public Sector Enterprises.

Markets - Perfect and Imperfect Competition - Features of Perfect Competition Monopoly- Monopolistic

Competition–Oligopoly-Price-Output Determination-Pricing Methods and Strategies

UNIT-IV

Capital Budgeting:. Working Capital meaning- significance - types, operating cycle- determinants, Capital Budgeting–Features, Proposals, Methods and Evaluation. Projects– Pay Back Method, Accounting Rate of Return (ARR) Net Present Value (NPV) Internal Rate Return (IRR) Method (sample problems)

UNIT-V

Financial Accounting and Analysis: Introduction – Nature, meaning, significance, functions and

advantages. Concepts and Conventions-Double-Entry Book Keeping, Journal, Ledger, Trial Balance-Final Accounts (Trading Account, Profit and Loss Account and Balance Sheet with simple adjustments).Financial Analysis-Analysis and Interpretation of Liquidity Ratios, Activity Ratios, and Capital structure Ratios and Profitability.

Textbooks:

1. Varshney &Maheswari: Managerial Economics, Sultan Chand,2013.

Reference Books:

2. Managerial Economics: Principles And Worldwide Applications, 9E (Adaptation) by Dominick Salvatore

and Siddhartha Rastogi

3. Managerial Economics: Principles and Worldwide Applications by Dominick Salvatore

CO-PO MAPPING:

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

Year & Sem	**III Year – I Semester**

Course Code	22UCS33	L	T	P	C
Course Name	DATA WAREHOUSING AND DATA MINING				

UNIT-1

Data Warehousing and Online Analytical Processing: Data Warehouse: Basic concepts, Data Warehouse Modelling: Data Cube and OLAP, Data Warehouse Design and Usage, Data Warehouse Implementation, Introduction: Why and What is data mining, What kinds of data need to be mined and patterns can be mined, Which technologies are used, Which kinds of applications are targeted.

UNIT-2

Data Pre-processing: An Overview, Data Cleaning, Data Integration, Data Reduction, Data Transformation and Data Discretization.

UNIT-3

Classification: Basic Concepts, General Approach to solving a classification problem, Decision Tree Induction: Attribute Selection Measures, Tree Pruning, Scalability and Decision Tree Induction, Visual Mining for Decision Tree Induction.

UNIT-4

Association Analysis: Problem Definition, Frequent Item set Generation, Rule Generation: Confident Based Pruning, Rule Generation in Apriori Algorithm, Compact Representation of frequent item sets, FPGrowth Algorithm.

UNIT-5

Cluster Analysis: Overview, Basics and Importance of Cluster Analysis, Clustering techniques, Different Types of Clusters; K-means: The Basic K-means Algorithm, K-means Additional Issues, Bi-secting K Means.

TEXT BOOKS:

1. Data Mining concepts and Techniques, 3/e, Jiawei Han, Michel Kamber, Elsevier, 2011.

2. Introduction to Data Mining: Pang-Ning Tan & Michael Steinbach, Vipin Kumar, Pearson, 2012.

REFERENCE BOOKS:

1. Data Mining Techniques and Applications: An Introduction, Hongbo Du, Cengage Learning.

2. Data Mining: VikramPudi and P. Radha Krishna, Oxford Publisher.

3. Data Mining and Analysis – Fundamental Concepts and Algorithms; Mohammed J. Zaki, Wagner Meira, Jr, Oxford

4. Data Warehousing Data Mining & OLAP, Alex Berson, Stephen Smith, TMH. http://onlinecourses.nptel.ac.in

5. (NPTEL course by Prof.Pabitra Mitra) http://onlinecourses.nptel.ac.in

6. (NPTEL course by Dr. Nandan Sudarshanam& Dr. Balaraman Ravindran) http://www.saedsayad.com

CO-PO MAPPING:

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

Year & Sem	III Year – I Semester				
Course Code	22UCS51	L	T	P	C
Course Name	FULL STACK TECHNOLOGIES				

Course Objectives:

- This course is designed to introduce students with no programming experience to the programming languages and techniques associated with the World Wide Web. The course will introduce web-based media-rich programming tools for creating interactive web pages.

Course Outcomes:

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

- Outline the concepts of Extensible markup language & AJAX
- Create and deploy secure, usable database driven web applications using Servlets, JSP.
- Create and deploy secure, usable database driven web applications using PHP .
- Build and developwebApplicationsusingPERL
- DevelopwebApplicationsusingRuby andframeworks

UNIT-I:

AJAX A New Approach: Introduction to AJAX, Why and What AJAX, Advantages of AJAX,AJAX Browser Support, How AJAX Works, Asynchronous (AJAX Web-Application Model), AJAX Technologies, AJAX-The XMLHttpRequest Object, AJAX-Send a **Request** to a Server, AJAX-server **Response,** AJAX XML File, Integrating PHP and AJAX.

UNIT-II:

Java Web Technologies–Servlets: Introduction to Servlet, Life cycle of Servlet, The Servlet API, The javax.servlet Package, Reading Servlet Parameters, The javax.servlet.http Package,Handling HTTP Requests and Responses, Using Cookies, Session Tracking

Java Server Pages(JSP): The problem with Servlets,Introduction to Java Server Pages, Anatomy of a JSP page, JSP Processing, JSP Elements, JSP with MySQL or Oracle

UNIT-III:

PHP Programming: Introduction to PHP, Creating PHP script, Running PHP script. Working with variables and constants: Using variables, Using constants, Data types, Operators. Controlling program flow: Conditional statements, Control statements, Arrays, functions, Working with forms and Databases such as MySQL.

UNIT-IV:

PERL Programming: Introduction to PERL. Operators and if statements, Program design and control structures, Arrays, Hashes and File handling, Regular expressions, Subroutines, Retrieving documents from the web with Perl.

UNIT-V:

Web Servers- IIS (XAMPP, LAMP) and Tomcat Servers.

Web development framework – Introduction to Ruby, Variables, Types, Simple I/O,Ruby Operators , ControlStatements, Arrays, Hashes, Methods, Classes, Iterators .

Text Books:

- Programming the World Wide Web, Robet W Sebesta, 7ed, Pearson.
-
- Web Technologies, Uttam K Roy, Oxford
- The complete Reference Java, 8th edition, Herbert Schildt, TMH.
- The Web Warrior Guide to Web Programming, Bai, Ekedahl, Farrelll, Gosselin, Zak, Karparhi, Maclntyre, Morrissey, Cengage

Reference Books

- Ruby on Rails Up and Running, Lightning fast Web development, Bruce Tate, Curt Hibbs, Oreilly (2006)

- Programming Perl, 4ed, Tom Christiansen, Jonathan Orwant, Oreilly (2012)

- Web Technologies, HTML< JavaScript, PHP, Java, JSP, XML and AJAX, Black book, Dream Tech.
- An Introduction to Web Design, Programming, Paul S Wang, Sanda S Katila, Cengage

CO-PO MAPPING:

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

Year & Sem	III Year – I Semester				
Course Code	22UCS34	L	T	P	C
Course Name	Artificial Intelligence				

CourseObjectives:

Thestudentwillbeableto

- KnowthemethodologyofProblem solving
- ImplementbasicAI algorithms
- Designandcarryoutanempiricalevolutionofdifferentalgorithms onaproblemformalization

CourseOutcomes (COs):

Attheendofthecourse,studentwillbeableto

- UnderstandthefundamentalconceptsinArtificialIntelligence
- Analyzetheapplications ofsearchstrategies andproblem reductions
- Applythemathematicallogicconcepts.
- DeveloptheKnowledgerepresentationsinArtificialIntelligence.
- ExplaintheFuzzylogicsystems.

UNITI:

Introduction to artificial intelligence: Introduction ,history, intelligent systems, foundations ofAI, applications, tic-tac-tiegameplaying,developmentofAIlanguages,currenttrendsinAI.

UNIT II:

Problemsolving:state-spacesearchandcontrolstrategies: Introduction,generalproblemsolving, characteristics ofproblem

SearchStrategies:exhaustivesearches,heuristicsearchtechniques,iterative-deepeningA*,constraint satisfaction

UNIT III:

Logic concepts: Introduction, propositional calculus, proportional logic, natural deduction system, axiomatic system,semantictableau systeminproportionallogic,predicatelogic

UNIT IV:

Knowledge representation: Introduction, approaches to knowledge representation, knowledge representationusingsemanticnetwork, extended semanticnetworksforKR

Advancedknowledgerepresentationtechniques: Introduction,conceptualdependencytheory,script structure.

UNIT V:

Expertsystemandapplications: Introductionphasesinbuildingexpertsystems,expertsystemversus traditionalsystems,rule-basedexpertsystemsblackboardsystemstruthmaintenancesystems,application

ofexpertsystems,listofshells andtools.

TextBooks:

1. ArtificialIntelligence, Rich,KevinKnight, ShivShankarB Nair,3rded,TMH
2. IntroductiontoArtificialIntelligence,Patterson,PHI

ReferenceBooks:

1. Artificialintelligence,structures andStrategiesforComplexproblem solving,-GeorgeFLugar,5th ed,PEA
2. IntroductiontoArtificialIntelligence,Ertel,WolfGang,Springer
3. Artificial Intelligence,AnewSynthesis, Nils J Nilsson,Elsevier
4. ArtificialIntelligence,SarojKaushik,CENGAGELearning,
5. Artificialintelligence,AmodernApproach,2nded,StuartRussel,PeterNorvig,PEA

CO-PO MAPPING:

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

Year & Sem	**III Year – I Semester**				
Course Code	**22UCS38**	**L**	**T**	**P**	**C**
Course Name	**DATA WAREHOUSING AND DATA MINING LAB**				

Course Objectives: The main objective of the course is to

x Inculcate Conceptual, Logical, and Physical design of Data Warehouses OLAP applications and

OLAP deployment

x Design a data warehouse or data mart to present information needed by management in a form that

is usable

x Emphasize hands-on experience working with all real data sets.

x Test real data sets using popular data mining tools such as WEKA, Python Libraries

x Develop ability to design various algorithms based on data mining tools.

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

x Design a data mart or data warehouse for any organization

x Extract knowledge using data mining techniques and enlist various algorithms used in information

analysis of Data Mining Techniques

x Demonstrate the working of algorithms for data mining tasks such as association rule mining,

classification for realistic data

x Implement and Analyze on knowledge flow application on data sets and Apply the suitable

visualization techniques to output analytical results

Software Requirements: WEKA Tool/Python/R-Tool/Rapid Tool/Oracle Data mining

List of Experiments:

1. Creation of a Data Warehouse.

¾ Build Data Warehouse/Data Mart (using open source tools like Pentaho Data Integration Tool,

Pentaho Business Analytics; or other data warehouse tools like Microsoft-SSIS, Informatica,

Business Objects,etc.,)

¾ Design multi-dimensional data models namely Star, Snowflake and Fact Constellation schemas for

any one enterprise (ex. Banking, Insurance, Finance, Healthcare, manufacturing, Automobiles,

sales etc).

¾ Write ETL scripts and implement using data warehouse tools.

¾ Perform Various OLAP operations such slice, dice, roll up, drill up and pivot

2. Explore machine learning tool “WEKA”

¾ Explore WEKA Data Mining/Machine Learning Toolkit.

¾ Downloading and/or installation of WEKA data mining toolkit.

¾ Understand the features of WEKA toolkit such as Explorer, Knowledge Flow interface,

Experimenter, command-line interface.

¾ Navigate the options available in the WEKA (ex. Select attributes panel, Preprocess panel, Classify

panel, Cluster panel, Associate panel and Visualize panel)

¾ Study the arff file format Explore the available data sets in WEKA. Load a data set (ex. Weather

dataset, Iris dataset, etc.)

¾ Load each dataset and observe the following:

1. List the attribute names and they types

2. Number of records in each dataset

3. Identify the class attribute (if any)

4. Plot Histogram

5. Determine the number of records for each class.

6. Visualize the data in various dimensions

3. Perform data preprocessing tasks and Demonstrate performing association rule mining on data sets

¾ Explore various options available in Weka for preprocessing data and apply Unsupervised filters

like Discretization, Resample filter, etc. on each dataset

¾ Load weather. nominal, Iris, Glass datasets into Weka and run Apriori

Algorithm with different support and confidence values.

¾ Study the rules generated. Apply different discretization filters on numerical attributes and run the

Apriori association rule algorithm. Study the rules generated.

¾ Derive interesting insights and observe the effect of discretization in the rule generation process.

4. Demonstrate performing classification on data sets

¾ Load each dataset into Weka and run 1d3, J48 classification algorithm. Study the classifier output.

Compute entropy values, Kappa statistic.

¾ Extract if-then rules from the decision tree generated by the classifier, Observe the confusion matrix.

¾ Load each dataset into Weka and perform Naïve-bayes classification and k-Nearest Neighbour

classification. Interpret the results obtained.

¾ Plot RoC Curves

¾ Compare classification results of ID3, J48, Naïve-Bayes and k-NN classifiers for each dataset, and

deduce which classifier is performing best and poor for each dataset and justify.

5. Demonstrate performing clustering of data sets

¾ Load each dataset into Weka and run simple k-means clustering algorithm with different values of

k (number of desired clusters).

¾ Study the clusters formed. Observe the sum of squared errors and centroids, and derive insights.

¾ Explore other clustering techniques available in Weka.

¾ Explore visualization features of Weka to visualize the clusters. Derive interesting insights and

explain.

6. Demonstrate knowledge flow application on data sets

¾ Develop a knowledge flow layout for finding strong association rules by using Apriori, FP Growth

algorithms

¾ Set up the knowledge flow to load an ARFF (batch mode) and perform a cross validation using J48

algorithm

¾ Demonstrate plotting multiple ROC curves in the same plot window by using j48 and Random

forest tree

7. Demonstrate ZeroR technique on Iris dataset (by using necessary preprocessing technique(s)) and

share your observations

8. Write a java program to prepare a simulated data set with unique instances.

9. Write a Python program to generate frequent item sets / association rules using Apriori algorithm

10. Write a program to calculate chi-square value using Python. Report your observation.

11. Write a program of Naive Bayesian classification using Python programming language.

12. Implement a Java program to perform Apriori algorithm

13. Write a program to cluster your choice of data using simple k-means algorithm using JDK

14. Write a program of cluster analysis using simple k-means algorithm Python programming language.

15. Write a program to compute/display dissimilarity matrix (for your own dataset containing at least four

instances with two attributes) using Python

16. Visualize the datasets using matplotlib in python.(Histogram, Box plot, Bar chart, Pie chart etc.,)

CO-PO MAPPING:

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

Year & Sem	**III Year – I Semester**				
Course Code	**22UCS39**	**L**	**T**	**P**	**C**
Course Name	**FULL STACK TECHNOLOGIES LAB**				

Course Objectives:

From the course the student will

- Learn the core concepts of both the frontend and backend programming course
- Get familiar with the latest web development technologies
- Learn all about PHP and SQL databases
- Ability to develop dynamic web content using Java Servlets andJSP
- To understand JDBC connections and Java MailAPI
- To understand the design and development process of a complete webapplication

Course Outcomes:

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

- Apply Web Application Terminologies, Internet Tools, E – Commerce and other web services
- Develop and Analyze dynamic Web Applications using PHP & MySql
- Install & Use Frameworks

- Develop Dynamic web content using Java Servlets andJSP
- Develop Dynamic web content using Perl andRuby
- To develop JDBC connections and implement a complete Dynamicweb application

1) Create Web pages using AJAX

2) Example program for Integrating PHP and AJAX

3) Implement a Servlet program on request response processing.

4) Implement a Servlet program for Registration Page.

5) Connect to a database using JSP and practice SQL Queries (MySql or Oracle).

6) Example PHP program for contact uspage.

7)UserAuthentication: Assume four users user1, user2, user3 and user4 having the passwords pwd1,

pwd2, pwd3 and pwd4 respectively. Write a PHP for doing the following.

i. Create a Cookie and add these four user id’s and passwords to thisCookie.

ii. Read the user id and passwords entered in the Login formand authenticate with the

values (user id and passwords) available in thecookies. If he is a valid user

(i.e., user-name and password match) you should welcome him by name (user-name) else you should display “You are not an authenticated user ’’.Use init-parameters to do this.

8)Example PHP program for registering users of a website andlogin.

9) Install a database (Mysql or Oracle). Create a table which should contain at least the following

fields: name, password, emailid, phone number (these should hold the data from the registration form). Write a PHP program to connect to that database and extract data from the tables and display them. Experiment with various SQL queries. Insert the details of the users who register with the web site, whenever a new user clicks the submit button in the registration.

10) Write a PHP which does the followingjob:

Insert the details of the 3 or 4 users who register with the web site by using registration form. Authenticate the user when he submits the login form using the user name and password from the database using cookies

11) Write a program which illustrates the use of associative arrays inperl.

12) Write perl program takes set names along the command line and prints whether theyare regular files or specialfiles

13) An example perl program to connect to a MySQl database table and executing simple commands.

14) Write a Ruby program which counts number of lines in a text

files using its regular Expressionsfacility.

15) Write programs which uses associative arrays concept ofRuby.

16) Define complex class in Ruby and do write methods to carry operations on complexobjects.

CO-PO MAPPING:

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

Year & Sem	**III Year – I Semester**				
Course Code	**22UCS40**	**L**	**T**	**P**	**C**
Course Name	**ANIMATION COURSE: ANIMATION DESIGN** **(Skill Oriented Course)**				

Course Objectives:

The main objective of this course is to understand 2-D and 3-D animation using Adobe package

Course Outcomes:

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

- learn various tools of digital 2-D animation
- understand production pipeline to create 2-D animation.
- apply the tools to create 2D animation for films and videos
- understand different styles and treatment of content in 3D model creation
- apply tools to create effective 3D modelling texturing and lighting
- **List of Experiments:**
- 1. Adobe Photoshop:

a. Create your visiting card

b. Create Title for any forthcoming film

2. Draw Cartoon Animation using reference

3. Creating Frame by Frame animation

4. Create a scene by using Mask layers animation

5. Adobe Illustrator: Packet Design(Toothpaste packet, Soap cover, any Food product)

6. Create any model of the male or female character

7. Create any Model of Cars or Bike

8. Create any Model of any animal.

9. Create any Model of any birds, fishes, and worms.

10. Create and Convert 2D objects into 3D objects

11. Create an animated 3D titling with sound

12. Create any Model some objects such as chairs, tables, fruits, utensils

List of Augmented Experiments: (Weeks 13 – Week 16)

(Any two of the following experiments can be performed)

1. Animate day and night scene of a street with the help of lighting

2. Create a human character using Character studio and animate the same

3. Create a natural outdoor or indoor scene.

4. Apply texture on various objects and characters

Reference Books:

1.Flash MX 2004, Thyagharajan Anbumani, TMH.

2.Brian Underdahl, The Complete Reference – Macromedia Flash Mx2004, 2nd edition – TMH.

Web Links:

1. https://onlinecourses.swayam2.ac.in/cec21_cs07/preview

2. https://onlinecourses.swayam2.ac.in/ugc19_cs09/preview

3. https://onlinecourses.swayam2.ac.in/ntr20_ed15/preview

4.https://youtube.com/playlist?list=PLfFk8y2fd3FjeE_CrFASNvDLBp3yF1Hwi

Year & Sem	**III Year – I Semester**				
Course Code	**22UEN15**	**L**	**T**	**P**	**C**
Course Name	**EMPLOYABILITY SKILLS-I**				

Introduction

A course on *Employability Skills-I* is considered essential at the third-year level of B.Tech. At this stage, the students need to prepare themselves for their career which requires them to listen to, read, speak and write in English both for their professional and interpersonal communication. The main purpose of this course is to prepare the students of Engineering for their placements.

Course Objectives: This ES-I focuses on using multi-media instruction for language development to meet the following targets:

- To improve students’ fluency in spoken English
- To enable them to listen to English spoken at normal conversational speed
- To help students develop their vocabulary
- To read and comprehend texts in different contexts
- To communicate their ideas relevantly and coherently in writing
- To make students industry-ready
- To respond appropriately in different socio-cultural and professional contexts

Learning Outcomes: Students will be able to:

- Acquire vocabulary and use it contextually

- Listen and speak effectively

- Develop proficiency in academic reading and writing

- Increase possibilities of job prospects Communicate confidently in formal and informal

contexts

Syllabus

1. **Inter-personal Communication and Vocabulary Building** - Starting a Conversation –

Responding Appropriately and Relevantly – Role Play in Different Situations -

Synonyms and Antonyms, One- word Substitutes, Prefixes and Suffixes, Idioms and

Phrases and Collocations.

2. **Reading Comprehension and Listening Skills** –General Vs Local Comprehension,

Techniques- Reading for Facts, Guessing Meanings from Context, Skimming, Scanning,

Inferring Meaning-Listening Comprehension (Video/Audio talks)

3. **Technical Writing Skills and Verbal ability** –Resume Writing – Tenses, Subject and

verb agreement, Voice, Correction of sentences, Spotting Errors and Sentence

Completion,

4. **Presentation Skills** &**Reasoning skills** - JAM ,Contextual vocabulary , Odd

man Out, Para jumbled sentences,

5. **Getting Ready for the Job:**

Group Discussion and Interview Skills – Dynamics of Group Discussion,

Intervention, Summarizing, Modulation of Voice, Body Language,

Relevance, Fluency and Organization of Ideas and Rubrics of Evaluation

Concept and Process, Pre-interview Planning, Opening Strategies, Answering Strategies, Interview through Tele-conference & Video conference and Mock Interviews.

Minimum Hardware Requirement:

Employability skills -1 shall have the following infrastructural facilities to accommodate at least 66 students in the hall

•**Spaciousroom with appropriate acoustics**

•**11 round tables with six movable chairsfor each table.**

•**Audio-visual aids**

•**LCD Projector**

•**Public Address system**

•**Computer with suitable configuration**

Suggested Software: The software consisting of the prescribed topics elaborated above should be procured and used.

•**Oxford Advanced Learner's Compass**, 10th Edition.

•**DELTA's key to the Next Generation TOEFL Test: Advanced Skill Practice.**

•**TOEFL & GRE(KAPLAN, AARCO & BARRONS, USA, CRACKING GRE by CLIFFS)**

•**TRAIN2SUCCESS.COM**

Suggested Reading:

1. English Vocabulary in Use series, CambridgeUniversityPress2008.

2. R.S.Aggarwal, A Modern Approach to Verbal & Non-Verbal Reasoning, S.Chand & CompanyLtd., 2018.

3. Barun K. Mitra, Personality Development and Soft Skills, Oxford University Press, 2011.

4. S.P. Dhanavel, English and Soft Skills, Orient Blackswan, 2010.

5. Raman, Meenakshi & Sharma, Sangeeta, Technical Communication

Principles and Practice,Oxford University Press, 2011.

5. Handbook for Technical Communication by David A.McMurrey

& Joanne Buckley.2012.

6. Handbook for Technical Writing by David A McMurrey &

JoanneBuckely CENGAGE Learning 2008.

7. Job Hunting by Colm Downes, Cambridge University Press2008.

8. Master Public Speaking by Anne Nicholls, JAICOPublishingHouse,2006.

9. English for Technical Communication for Engineering

Students, Aysha Vishwamohan, Tata Mc Graw-Hil 2009.10. Books on TOEFL/GRE/GMAT/CAT/IELTS/SAT by

Barron's/DELTA/Cambridge University Press.

11. The Definitive Book of body Language – by Allan Pease, Barbara Pease.

e-resources:

www. Indiabix.com

www.freshersworld.com

Year &Sem	**III Year – II Semester**				
Course Code	**22UCS41**	**L**	**T**	**P**	**C**
		3	**0**	**0**	**3**
Course Name	**Machine Learning**				

Course Objectives: The student should be able to:

1. Recognize the importance and characteristics of machine learning.
2. Apply supervised machine learning techniques for data handling and to gain knowledge from it.
3. Apply advanced supervised machine learning and probabilistic models for classification problems.
4. Apply unsupervised machine learning models to real world problems.
5. Evaluate the performance of algorithms and to provide solution for various real-world applications using ensemble models.

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

CO1: Recognize the characteristics of machine learning.

{Understand level, KL2} {Analyze level, KL4}

CO2: Apply various supervised learning methods to appropriate problems.

{Understand level, KL2} {Apply level, KL3}

CO3: Identify and integrate more than one technique to enhance the performance of learning and create probabilistic models for handling unknown pattern.

{Understand level, KL2} {Apply level, KL3} {Evaluate level, KL5}

CO4: Apply unsupervised learning models e.g. clustering algorithms to handle the unknown data.

{Apply level, KL3} {Analyze level, KL4}

CO5: Apply Ensemble models to any real-world problem to Analyze its performance effectively.

{Apply level, KL3} {Analyze level, KL4} {Evaluate level, KL5}

UNIT-I

Introduction to Machine Learning

Introduction, Components of Learning, Learning Models, Geometric Models, Probabilistic Models, LogicModels, Grouping and Grading, Designing a Learning System, Types of Learning, Supervised,Unsupervised, Reinforcement, Perspectives and Issues, Version Spaces, PAC Learning, VC Dimension.

UNIT-II

Supervised Learning

Learning a Class from Examples, Linear, Non-linear, Multi-class and Multi-label classification, DecisionTrees: ID3, Classification and Regression Trees (CART), Regression: Linear Regression, Multiple LinearRegression, Logistic Regression.

UNIT-III

Advanced Supervised Learning

Neural Networks: Introduction, Perceptron, Multilayer Perceptron, Support vector machines: Linear andNon-Linear, Kernel Functions, K-Nearest Neighbors.

Probabilistic Models Bayesian Learning, Bayes Optimal Classifier, Naïve Bayes Classifier, Bayesian Belief Networks.

UNIT– IV

Unsupervised Learning

Introduction to clustering, K-means clustering, K-Mode Clustering, Distance based clustering, Clusteringaround mediods, Silhouettes, Hierarchical Clustering.

UNIT-V:

Ensemble Learning

Ensemble Learning Model Combination Schemes, Voting, Error-Correcting Output Codes, Bagging:RandomForest Trees, Boosting: Adaboost, Stacking.

Text Books:

1. EthemAlpaydin,"Introduction to Machine Learning", MIT Press, Prentice Hall of India, Third Edition 2014.
2. MehryarMohri, AfshinRostamizadeh, AmeetTalwalkar "Foundations of Machine Learning", MIT Press, 2012
3. Machine Learning: The art and Science of algorithms that make sense of data, Peter Flach, Cambridge University Press, 2012

Reference Books:

1. Chris Albon: Machine Learning with Python Cookbook, O'Reilly Media, Inc.2018.
2. Tom Mitchell, "Machine Learning", McGraw Hill, 3rd Edition, 1997.
3. Charu C. Aggarwal, "Data Classification Algorithms and Applications", CRC Press, 2014. Stephen Marsland, "Machine Learning – An Algorithmic Perspective", 2nd Edition, CRC Press, 2015.
4. Kevin P. Murphy "Machine Learning: A Probabilistic Perspective", The MIT Press, 2012 Jiawei Han and Micheline Kambers and Jian Pei, "Data Mining –Concepts and Techniques", 3rd Edition, Morgan Kaufman Publications, 2012.

5. Marc Peter Deisenroth, A. Aldo Faisal, Cheng Soon Ong, “Mathematics for Machine Learning”, Cambridge University Press, 2019.

E- Resources & Other Digital Material:

1. Kevin Murphy, “Machine Learning: A Probabilistic Perspective”, MIT Press, 2012,

https://www.cs.ubc.ca/~murphyk/MLbook/pml-intro-5nov11.pdf.

2. Professor S. Sarkar, IIT Kharagpur “Introduction to machine learning”,

https://www.youtube.com/playlist?list=PLYihddLFCgYuWNL55Wg8ALkm6u8U7gps.

3. Professor Carl Gustaf Jansson, KTH, Video Course on Machine Learning

https://nptel.ac.in/noc/individual_course.php?id=noc19-cs35.

4. Tom Mitchell, “Machine Learning”,
http://www.cs.cmu.edu/~tom/10701_sp11/lectures.shtml

CO-PO MAPPING:

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

Year &Sem	**III Year – II Semester**				
Course Code	**22UCS42**	**L**	**T**	**P**	**C**
		3	**0**	**0**	**3**
Course Name	**Advanced Data Structures and Algorithms**				

Course Objectives:

The main objectives of the course is to

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

UNIT – I:

Introduction to Algorithm Analysis, Space and Time Complexity analysis, Asymptotic Notations. AVL Trees – Creation, Insertion, Deletion operations and Applications

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

UNIT – II:

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

Graphs – Terminology, Representations, Basic Search and Traversals, Connected

Components and Biconnected Components, applications

Divide and Conquer: The General Method, Quick Sort, Merge Sort, Strassen’s matrix

multiplication, Convex Hull

UNIT – III:

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

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

UNIT – IV:

Backtracking: General Method, 8-Queens Problem, Sum of Subsets problem, Graph

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

UNIT – V:

NP Hard and NP Complete Problems: Basic Concepts, Cook's theorem

NP Hard Graph Problems: Clique Decision Problem (CDP), Chromatic Number Decision Problem (CNDP), Traveling Salesperson Decision Problem (TSP)

NP Hard Scheduling Problems: Scheduling Identical Processors, Job Shop Scheduling

Textbooks:

1. Fundamentals of Data Structures in C++, Horowitz, Ellis; Sahni, Sartaj; Mehta,

Dinesh, 2ndEdition Universities Press

2. Computer Algorithms in C++, Ellis Horowitz, SartajSahni, SanguthevarRajasekaran,

2 nd Edition University Press

Reference Books:

1. Data Structures and program design in C, Robert Kruse, Pearson Education Asia

2. An introduction to Data Structures with applications, Trembley& Sorenson, McGraw Hill

3. The Art of Computer Programming, Vol.1: Fundamental Algorithms, Donald E Knuth, Addison-Wesley, 1997.

4. Data Structures using C & C++: Langsam, Augenstein&Tanenbaum, Pearson, 1995

5. Algorithms + Data Structures & Programs:, N.Wirth, PHI

6. Fundamentals of Data Structures in C++: Horowitz Sahni& Mehta, Galgottia Pub.

7. Data structures in Java:, Thomas Standish, Pearson Education Asia

Online Learning Resources:

1. https://www.tutorialspoint.com/advanced_data_structures/index.asp

2. http://peterindia.net/Algorithms.html

3. Abdul Bari,Introduction to Algorithms (youtube.com)

CO-PO Mapping

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

Year &Sem	III Year – II Semester				
Course Code	22UCS43	L	T	P	C
		3	0	0	3
Course Name	CRYPTOGRAPHY AND NETWORK SECURITY				

Course Objectives:

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

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

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

UNIT I:

Basic Principles : Security Goals, Cryptographic Attacks, Services and Mechanisms, Mathematics of Cryptography.

UNIT II:

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

UNIT III:

Asymmetric Encryption: Mathematics of Asymmetric Key Cryptography, Asymmetric KeyCryptography

UNIT IV:

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

UNIT V:

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

Text Books:

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

Reference Books:

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

CO-PO MAPPING:

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

Year &Sem	III Year – II Semester				
Course Code	22UCS46	L	T	P	C
		3	0	0	3
Course Name	OBJECT ORIENTED ANALYSIS & DESIGN				

Course Objectives:

1.The main objective is the students to Become familiar with all phases of OOAD. Master the main features of the UML.

2. Master the main concepts of Object Technologies and how to apply them at work and develop the ability to analyze and solve challenging problem in various domains.

3. Learn the Object design Principles and understand how to apply them towards Implementation.

Course Outcomes:

1. After finishing this course student will be able to: Analyze the nature of complex system and its solutions.

2.Illustrate & relate the conceptual model of the UML, identify & design the classes and relationships

3. Analyze &Design Class and Object Diagrams that represent Static Aspects of a Software System and apply basic and Advanced Structural Modeling Concepts for designing real time applications.

4. Analyze & Design behavioral aspects of a Software System using Use Case, Interaction and Activity Diagrams.

5. Analyze & Apply techniques of State Chart Diagrams and Implementation Diagrams to model behavioral aspects and Runtime environment of Software Systems.

UNIT I:

Introduction: The Structure of Complex systems, The Inherent Complexity of Software, Attributes of Complex System, Organized and Disorganized Complexity, Bringing Order to Chaos, Designing Complex Systems.

Case Study: System Architecture: Satellite-Based Navigation

UNIT II:

Introduction to UML: Importance of modeling, principles of modeling, object oriented modeling, conceptual model of the UML, Architecture, and Software Development Life Cycle.

Basic Structural Modeling: Classes, Relationships, common Mechanisms, and diagrams.

Case Study: Control System: Traffic Management.

UNIT III:

Class & Object Diagrams: Terms, concepts, modeling techniques for Class & Object Diagrams.

Advanced Structural Modeling: Advanced classes, advanced relationships, Interfaces, Types and Roles,
Packages. **Case Study:** AI: Cryptanalysis.

UNIT IV:

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

UNIT V:

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

Case Study: Weather Forecasting

Text Books:

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

Reference Books:

1. Meilir Page-Jones: Fundamentals of Object Oriented Design in UML, Pearson Education.
2. Pascal Roques: Modeling Software Systems Using UML2, WILEY- Dreamtech India Pvt. Ltd.
3. Atul Kahate: Object Oriented Analysis & Design, The McGraw-Hill Companies.
4. Appling UML and Patterns: An introduction to Object – Oriented Analysis and Design and Unified Process, Craig Larman, Pearson Education.

CO-PO MAPPING:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO.1	3	2	2	1	1	-	-	-	1	1	1	2
CO.2	3	3	3	3	2	2	1	1	1	1	1	2
CO.3	3	3	3	2	2	1	1	1	1	1	1	2
CO.4	3	3	2	2	2	2	1	1	1	1	1	2
CO.5	3	3	3	2	1	2	2	1	1	1	1	2
MEAN	1	0.933333	0.866667	0.666667	0.533333	0.466667	1	1	0.333333	0.333333	0.333333	0.666667

Year &Sem	**III Year – II Semester**				
Course Code	**22UCS53**	**L**	**T**	**P**	**C**
		3	**0**	**0**	**3**
Course Name	**MEAN STACK DEVELOPMENT**				

Course Objectives:

From the course the student will learn

- Translate user requirements into the overall architecture and implementation of new systems and Manage Project and coordinate with the Client
- Understand, create and debug database related queries and Create test code to validate the applications against client requirement
- Monitor the performance of web applications & infrastructure and Troubleshooting web application with a fast and accurate a resolution

Course Outcomes:

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

- Apply JavaScript to embed programming interface for web pages and also to perform Client side validations.
- Utilize Angular to design dynamic web pages and build a basic web server using Node.js work with Node Package Manager (NPM).
- Build a web server using Express.js
- Develop JavaScript applications using Typescript
- Develop web applications using scripting languages and frameworks and work with document database using MongoDB.

UNIT-I:

JavaScript: Introduction, Working with Identifiers, Type of Identifiers, Primitive and Non Primitive Data Types, Operators and Types of Operators, Types of Statements, Non - Conditional Statements, Types of Conditional Statements, If and Switch Statements, Types of Loops, Types of Functions, Declaring and Invoking Function, Arrow Function, Built-in Functions, Working with Classes, Creating and Inheriting Classes, Working with Objects, Types of Objects, Creating Objects, Creating Arrays, Accessing Arrays, Array Methods, Creating and Consuming Modules.

UNIT-II:

Node.js: Why and What Node.js, Install Node.js, Create a web server in Node.js, Node Package Manager, Modular programming in Node.js, Restarting Node Application, File Operations.

Express.js: Introduction, Defining a route, Handling Routes, How Middleware works, Chaining of Middlewares ,Types of Middlewares, Connecting to MongoDB with Mongoose, Models, CRUD Operations, Why Session management, Cookies, Sessions.

UNIT-III:

Typescript: Installing Typescript, Basics of Typescript, Function, Parameter Types and Return Types, Arrow Function, Optional and Default Parameters, Rest Parameter, Classes, Objects, Constructor, Abstract class, Inheritance, Access Modifiers, Properties and Methods, Creating an Interface, Extending Interfaces, Duck Typing, Function Types in Interface, Creating and using Namespaces, Creating and using Modules.

UNIT-IV:

MongoDB: Introduction to MongoDB, Installing MongoDB on the local computer (Windows or Mac), Introduction to MongoDB Cloud, Install and Configure MongoDB Compass, Create MongoDB Atlas Cluster, Introduction to the MongoDB Shell, MongoDB Shell JavaScript Engine, MongoDB Shell JavaScript Syntax, MongoDB Data Types, Introduction to the CRUD Operations on documents, Create and Delete Databases and Collections.

UNIT-V:

Angular: What is Angular, Features of Angular, difference between AngularJS and Angular, Angular Application Setup, Components and Modules, Executing Angular Application, Elements of Template, Structural Directives - ngIf, ngFor, ngSwitch, Attribute Directives- ngStyle, ngClass, Property Binding, Attribute Binding, Style and Event Binding, Built in Pipes, Passing Parameters to Pipes, Communicating with different backend services using Angular HttpClient.

Text Books:

1. Programming the World Wide Web, 7th Edition, Robet W Sebesta, Pearson.

2. Pro Mean Stack Development, 1st Edition, ELadElrom, Apress O’Reilly.

3. Full Stack JavaScript Development with MEAN, Colin J Ihrig, Adam Bretz, 1st edition,

   SitePoint, SitePoint Pty. Ltd., O'Reilly Media.

4. MongoDB – The Definitive Guide, 2nd Edition, Kristina Chodorow,O’Reilly.

5. Web Technologies, 1st Edition 7th impression, Uttam K Roy, Oxford, 2012.

Reference Books:

1. Web Technologies, HTML, JavaScript, PHP, Java, JSP, XML and AJAX, Black book,1st

   Edition, Dream Tech.

2. An Introduction to Web Design, Programming, 1st Edition, Paul S Wang, Sanda SKatila, Cengage Learning.

Web Links:

1. https://infyspringboard.onwingspan.com/en/app/toc/lex_18109698366332810000_shared/overview (Javascript)
2. https://infyspringboard.onwingspan.com/en/app/toc/lex_32407835671946760000_shared/overview (Node.js & Express.js)
3. https://infyspringboard.onwingspan.com/en/app/toc/lex_9436233116512678000_shared/overview (Typescript)
4. https://infyspringboard.onwingspan.com/en/app/toc/lex_20858515543254600000_shared/overview (Angular JS)
5. https://infyspringboard.onwingspan.com/en/app/toc/lex_auth_013177169294712832113_shared/overvie w(MongoDB)

CO-PO MAPPING

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO.1	3	3	2	2	2	1	1	1	1	1	1	2
CO.2	3	3	3	2	2	1	1	1	1	1	1	2
CO.3	3	3	3	2	2	1	1	1	1	1	1	2
CO.4	3	2	2	1	1	1	1	1	1	1	1	2
CO.5	3	3	3	2	2	1	1	1	1	1	1	2
MEAN	1	0.933	0.866	0.6	0.6	0.333	0.333	0.333	0.333	0.333	0.333	0.6667

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

Year &Sem	**III Year – II Semester**				
Course Code	**22UCS48**	**L**	**T**	**P**	**C**
		0	**0**	**3**	**1.5**
Course Name	**MACHINE LEARNING USING PYTHON LAB**				

Course Objectives:
This course will enable students to learn and understand different Data sets in implementing the machine learning algorithms.

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

- Implement procedures for the machine learning algorithms
- Design and Develop Python programs for various Learning algorithms
- Apply appropriate data sets to the Machine Learning algorithms
- Develop Machine Learning algorithms to solve real world problems

Requirements: Develop the following program using Anaconda/ Jupiter/ Spider and evaluate ML models.

Experiment-1:
Implement and demonstrate the FIND-S algorithm for finding the most specific hypothesis based on a given set of training data samples. Read the training data from a .CSV file.

Experiment-2:
For a given set of training data examples stored in a .CSV file, implement and demonstrate the Candidate-Elimination algorithm to output a description of the set of all hypotheses consistent with the training examples.

Experiment-3:
Write a program to demonstrate the working of the decision tree based ID3 algorithm. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample.

Experiment-4:
Exercises to solve the real-world problems using the following machine learning methods: a) Linear Regression b) Logistic Regression c) Binary Classifier

Experiment-5: Develop a program for Bias, Variance, Remove duplicates , Cross Validation

Experiment-6: Write a program to implement Categorical Encoding, One-hot Encoding

Experiment-7:
Build an Artificial Neural Network by implementing the Back propagation algorithm and test the same using appropriate data sets.

Experiment-8:
Write a program to implement k-Nearest Neighbor algorithm to classify the iris data set. Print both correct and wrong predictions.

Experiment-9: Implement the non-parametric Locally Weighted Regression algorithm in order to fit data points. Select appropriate data set for your experiment and draw graphs.

Experiment-10:

Assuming a set of documents that need to be classified, use the naïve Bayesian Classifier model to perform this task. Built-in Java classes/API can be used to write the program. Calculate the accuracy, precision, and recall for your data set.

Experiment-11: Apply EM algorithm to cluster a Heart Disease Data Set. Use the same data set for clustering using k-Means algorithm. Compare the results of these two algorithms and comment on the quality of clustering. You can add Java/Python ML library classes/API in the program.

Experiment-12: Exploratory Data Analysis for Classification using Pandas or Matplotlib.

Experiment-13:

Write a Python program to construct a Bayesian network considering medical data. Use this model to demonstrate the diagnosis of heart patients using standard Heart Disease Data Set

Experiment-14:

Write a program to Implement Support Vector Machines and Principle Component Analysis

Experiment-15:

Write a program to Implement Principle Component Analysis

CO-PO MAPPING:

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

Year &Sem	**III Year – II Semester**				
Course Code	**22UCS49**	**L**	**T**	**P**	**C**
		0	**0**	**3**	**1.5**
Course Name	**MEAN STACK DEVELOPMENT LAB**				

Course Objectives:

From the course the student will

- Learn the core concepts of both the frontend and backend programming course
- Get familiar with the latest web development technologies
- Learn all about SQL and Mongo databases.
- To understand the design and development process of a complete web application

Course Outcomes:

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

- Utilize JavaScript for developing interactive HTML web pages and validate form data.
- Build a basic web server using Node.js and also working with Node Package Manager (NPM).
- Develop and Analyze dynamic Web Applications using Express.js & MongoDB
- Make use of Typescript to optimize JavaScript code by using the concept of strict type checking.
- Develop and Analyze dynamic Web Applications using Angular.

Exercise -1 (JavaScript - Loops, Arrays)

a).Write a JavaScript code to book movie tickets online and calculate the total price based on the 3 conditions:

(i) If seats to be booked are not more than 2, the cost per ticket remains Rs. 150.

(ii) If seats are 6 or more, booking is not allowed.

(iii) If seats to be booked are more than 2 but less than 5, based on the number of seats booked. (festive season discount of 10%)

b). Write a JavaScript to create an array of objects having movie details. The object should include the movie name, starring, language, and ratings. Render the details of movies on the page using the array.

Exercise -2 (JavaScript - Inheritance, Modules)

a).Write a JavaScript to create an Employee class extending from a base class Person. Hints:

(i) Create a class Person with name and age as attributes.

(ii) Add a constructor to initialize the values

(iii) Create a class Employee extending Person with additional attributes role

b). Write a JavaScript to validate the user by creating a login module. Hints: (i) Create a file login.js with a User class. (ii) Create a validate method with username and password as arguments. (iii) If the username and password are equal it will return "Login Successful" else will return "Unauthorized access".

Exercise -3 (Node.js)

a).Write a program to show the workflow of JavaScript code executable by creating web server in Node.js.

b). Write a Node.js module to show the workflow of Modularization of Node application

c). Write a Node.js program to show the workflow of restarting a Node application.

Exercise -4 (Express.js-Middleware, MongoDB)

a).Implement Middleware for myNotes application: (i) we want to handle POST submissions. (ii) display customized error messages. (iii) perform logging.

b). Write a Express.js program to write a Mongoose schema to connect with MongoDB

Exercise -5 (Express.js –Mongoose, MongoDB)

a). Write a Express.js program to perform various CRUD (Create-Read-Update-Delete) operations using Mongoose library functions.

Exercise -6 (Express.js - Session Management)

a). Write a Express.js program to explain session management using cookies.

b). Write a Express.js program to explain session management using sessions.

Exercise - 7 (Typescript - Interfaces)

a).Write a Typescript program to declare an interface named - Product with two properties like productId and productName with a number and string data type and need to implement logic to populate the Product details using Duck typing.

b). Write a Typescript program to declare a productList interface which extends properties from two other declared interfaces like Category, Product as well as implementation to create a variable of this interface type.

Exercise -8 (Typescript – Classes, Access modifiers, Namespaces)

a). Write a Typescript program to create a Product class with 4 properties namely productId, productName, productPrice, productCategory with private, public, static, and protected access modifiers and accessing them through Gadget class and its methods.

b). Write a Typescript program to create a namespace called ProductUtility and place the Product class definition in it.Import the Product class inside productlist file and use it.

Exercise -9 (MongoDB)

a) Write MongoDB queries to Create and Delete databases and collections.

b) Write MongoDB queries to work with records using find(), limit(), sort(), update() and count().

Exercise -10 (Angular JS, Angular)

a) Write a Angular JS program to implements the concept of Form Validation and Form Submission

b).Write a Angular program to create a new component called hello and render Hello Angular on the page.

Web Links:

1. https://infyspringboard.onwingspan.com/en/app/toc/lex_18109698366332810000_shared/overview (Javascript)
2. https://infyspringboard.onwingspan.com/en/app/toc/lex_32407835671946760000_shared/overview (Node.js & Express.js)
3. https://infyspringboard.onwingspan.com/en/app/toc/lex_9436233116512678000_shared/overview (Typescript)
4. https://infyspringboard.onwingspan.com/en/app/toc/lex_20858515543254600000_shared/overview

(Angular JS)

5. https://infyspringboard.onwingspan.com/en/app/toc/lex_auth_013177169294712832113_shared/overview (MongoDB)

CO-PO MAPPING:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	2	2	2	1	1	1	1	1	1	2
CO2	3	2	3	3	2	1	1	1	1	1	1	2
CO3	3	3	3	2	2	1	1	1	1	1	1	2
CO4	3	2	2	1	1	1	1	1	1	1	1	2
CO5	3	3	3	2	2	1	1	1	1	1	1	2
MEAN	1	0.866667	0.866667	0.666667	0.6	0.333333	0.333333	0.333333	0.333333	0.333333	0.333333	0.66666

Year &Sem	III Year – II Semester				
Course Code	22UCS50	L	T	P	C
		0	0	3	1.5
Course Name	Advanced Data Structures and Algorithms Lab				

Course Objectives:

The objectives of the course is to

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

Experiments covering the Topics:

- Operations on AVL trees, B-Trees, Heap Trees
- Graph Traversals
- Sorting techniques
- Minimum cost spanning trees
- Shortest path algorithms
- 0/1 Knapsack Problem
- Travelling Salesperson problem
- Optimal Binary Search Trees
- N-Queens Problem
- Job Sequencing

Sample Programs:

1. Construct an AVL tree for a given set of elements which are stored in a file. And

implement insert and delete operation on the constructed tree. Write contents of tree

into a new file using in-order.

2. Construct B-Tree an order of 5 with a set of 100 random elements stored in array.

Implement searching, insertion and deletion operations.

3. Construct Min and Max Heap using arrays, delete any element and display the content of the Heap.

4. Implement BFT and DFT for given graph, when graph is represented by

a) Adjacency Matrix b) Adjacency Lists

5. Write a program for finding the biconnected components in a given graph.

6. Implement Quick sort and Merge sort and observe the execution time for various

input sizes (Average, Worst and Best cases).

7. Compare the performance of Single Source Shortest Paths using Greedy method when the graph is represented by adjacency matrix and adjacency lists.

8. Implement Job Sequencing with deadlines using Greedy strategy.

9. Write a program to solve 0/1 Knapsack problem Using Dynamic Programming.

10. Implement N-Queens Problem Using Backtracking.

11. Use Backtracking strategy to solve 0/1 Knapsack problem.

12. Implement Travelling Sales Person problem using Branch and Bound approach.

Reference Books:

1. Fundamentals of Data Structures in C++, Horowitz Ellis, SahniSartaj, Mehta, Dinesh, 2 ndEdition, Universities Press

2. Computer Algorithms/C++ Ellis Horowitz, SartajSahni, SanguthevarRajasekaran,

2 ndEdition, University Press

3. Data Structures and program design in C, Robert Kruse, Pearson Education Asia

4. An introduction to Data Structures with applications, Trembley& Sorenson, McGraw Hill

Online Learning Resources:

1. http://cse01-iiith.vlabs.ac.in/

2. http://peterindia.net/Algorithms.html

CO-PO Mapping

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

Year &Sem	III Year – II Semester				
Course Code	22UCS52	L	T	P	C
		0	1	2	2
Course Name	**ANDROID LAB**				

Course objectives: The main objectives are

1. To understand the installation of Android SDK
2. To explore various features of Android SDK tools
3. To explore various components in Android SDK

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

1. Explore features of Android SDK
2. Explore various components in GUI development through Android
3. Experiment different applications through Android SDK

List of Experiments

1. Procedure to install Android Studio.
2. Develop a sample “Hello World Application.
3. Create an application that takes the name from a text box and shows hello message along with the name entered in text box, when the user clicks the OK button.
4. Create a screen that has input boxes for User Name, Password, Address, Gender(radio buttons for male and female), Age (numeric), Date of Birth (Date Picket), State (Spinner) and a Submit button. On clicking the submit button, print all the data below the Submit Button (use any layout).
5. Design an android application to create page using Intent and one Button and pass the Values from one Activity to second Activity.
6. Design an android application Send SMS using Intent.
7. Create an android application using Fragments.
8. Design an android application for menu.
9. Design an android application to display list of fruits using Listview
10. Design an android application to display list of users using Recyclerview from internet.

CO-PO MAPPING:

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