Curriculum for UG Degree Course in

B.TECH. INFORMATION TECHNOLOGY

Regulation 2022 (Revised)

CHENNAI INSTITUTE OF TECHNOLOGY (Autonomous)

Document Version

Version Number	Date	Author	Major Updates	Approved by
1.0	24.4.2024	Dr. A.R. KAVITHA	Python Programming	
			Data Analytics and Visualization	
			Computer Architecture and Organization	
			Principles of Communication Systems	
			Cloud and Distributed Computing	
			Mobile Computing	

Section 1: General Course Structure

A. Definition of Credit:

1 Hour Lecture (L) per week	1 Credit
1 Hour Tutorial (T) per week	1 Credit
1 Hour Practical (P) per week	0.5 Credit

B. Structure of Program

S. No.	Category	Credits
1	Humanities & Social Science Courses (HSMC)	16
2	Basic Science Courses (BSC)	24
3	Engineering Science Courses (ESC)	16
4	Program Core Courses (PCC)	65
5	Professional Elective Courses (PEC)	18
6	Open Elective Courses (OEC)	6
7	Employability Enhancement Skills (EES)	23
8	Mandatory Course (MC)	-
	TOTAL	**168**

C. Course code and definition

Code	Definition
L	Lecture
T	Tutorial
P	Practical
C	Credits
IT	Professional core courses
IT PE	Professional Elective courses
IT OE	Open Elective Courses
IT MC	Mandatory Courses

Course level coding scheme: Four-digit number is used as a suffix with the Course Code for identifying the level of the course. Thousand's place denotes the regulation number (we use "3" for 2022-23 Regulation) Digit at hundred's place signifies the semester in which the course is offered. The last two digits represent the serial order of course within the semester. For example, 3101, 3102, … are courses offered during the first semester.

D. Category-wise Courses

Humanities & Social Science Courses (HSMC)

S. No.	Course Title	Semester	L	T	P	C
1.	தமிழர் மரபு /Heritage of Tamils	I	1	0	0	1
2.	Communicative English – I	I	3	0	2	4
3.	தமிழரும் தொழில் நுட்பமும் /Tamil and Technology	II	1	0	0	1
4.	Communicative English – II	II	3	0	2	4
5.	Managerial Economics & Financial Accounting	VI	3	0	0	3
6.	Professional Ethics & Human Values	VII	3	0	0	3
Total Credits			**16**			

Basic Science Courses (BSC)

S. No.	Course Title	Semester	L	T	P	C
1.	Matrices and Calculus	I	3	1	0	4
2.	Engineering Physics	I	3	0	2	4
3.	Engineering Chemistry	I	3	0	2	4
4.	Probability and statistics	II	3	1	0	4
5.	Linear Algebra	III	3	1	0	4
6.	Discrete Mathematics	IV	3	1	0	4
Total Credits			**24**			

Engineering Science Courses (ESC)

S. No.	Course Title	Semester	L	T	P	C
1.	Web Development Essentials	I	3	0	2	4
2.	Problem Solving Using C++ Programming	I	3	0	2	4
3.	Digital System Design	II	3	0	2	4
4.	Embedded Programming	V	3	0	2	4
Total Credits			**16**			

Program Core Courses (PCC)

S. No.	Course Title	Semester	L	T	P	C
1.	Data Structures using C++	II	3	0	2	4
2.	Database Management Systems	II	3	0	2	4
3.	Introduction to Java Programming	II	3	0	2	4
4.	Computer Architecture	III	3	0	0	3
5.	Design and Analysis of Algorithms	III	3	0	2	4
6.	Web Development Practices	III	3	0	2	4
7.	Computer Networks	III	3	0	2	4
8.	Python Programming	III	3	0	2	4

9.	Theory of Computation	IV	3	0	0	3
10.	Operating Systems	IV	3	0	2	4
11.	Introduction to Artificial Intelligence and Machine Learning	IV	3	0	2	4
12.	Introduction to Cyber Security	IV	3	0	2	4
13.	Software Engineering	IV	3	0	2	4
14.	Computing Theory and Compiler Design	V	3	0	2	4
15.	Data Analytics and Visualization	V	3	0	2	4
16.	Distributed Computing	VI	3	0	0	3
17.	Software Testing	VI	3	0	2	4
Total Credits			**65**			

Professional Elective courses (PEC)

S. No.	Course Title	Semester	L	T	P	C
1.	Professional Elective – I	V	2/3	0	2/0	3
2.	Professional Elective – II	V	2/3	0	2/0	3
3.	Professional Elective – III	VI	2/3	0	2/0	3
4.	Professional Elective – IV	VI	2/3	0	2/0	3
5.	Professional Elective – V	VII	2/3	0	2/0	3
6.	Professional Elective – VI	VII	2/3	0	2/0	3
Total Credits						**18**

Open Elective Courses (OEC)

S. No.	Course Title	Semester	L	T	P	C
1.	Open Elective – I	VI	3	0	0	3
2.	Open Elective – II	VII	3	0	0	3
Total Credits			**6**			

Mandatory Course (MC)

S. No.	Course Title	Semester	L	T	P	C
1.	Mandatory Course – I	III	2	0	0	0
2.	Mandatory Course – II	IV	2	0	0	0
Total Credits			**6**			

E. Induction Program

- *Catapult* is a dynamic week-long event designed for our incoming first-year students, offering an immersive introduction to the diverse array of clubs and activities across the college campus. In addition tofamiliarizing them with our labs and Centers of Excellence (COEs), Catapult aims to acclimate first-year students to college life, ensuring they feel at ease with the forthcoming experiences of their four-year journey.
- This initiative fosters meaningful connections between seniors and juniors, providing a platform for them to explore departmental projects and engage in

collaborative activities, thereby enhancing camaraderie and knowledge sharing within the college community.

F. Evaluation Scheme

a. For Theory Courses:
The weightage of Internal assessment is 40% and for End Semester Exam is 60%
The student has to obtain at least 50% marks individually both in internal assessment and end semester exams to pass

b. For Practical Courses:
The weightage of Internal assessment is 60% and for End Semester Exam is 40%
For Theory cum Lab
The student has to obtain at least 50% marks individually both in internal assessment and end semester exams to pass.

c. For Theory Cum Practical Courses:
The weightage of Internal assessment is 50% and for End Semester Exam is 50%
The student has to obtain at least 50% marks individually both in internal assessment and end semester exams to pass

Note: The internal assessment is based on the student's performance in 3 Internal Assessment (IA) exams, quizzes, assignments, class performance, attendance, etc.

d. For Project works:

Assessment of project works comprises three internal reviews and an end-of-semester evaluation. Internal reviews, worth 40 marks in total, encompass assessment criteria such as Project Synopsis/Proposal Evaluation, Methodology and Design of Existing System, Feasibility of Project Proposal, Planning of Project Work, and Team Work. At the conclusion of the semester, 20 marks are designated for assessing the quality of the report, while the remaining 40 marks are reserved for evaluating performance in viva-voce, demonstration of the work, and other relevant factors.

G. Learning Beyond Class Room
a. Students should be encouraged to visit Centers of Excellence (COEs) on campus and learn additional technical skills
b. Students should be encouraged to participate in internal/external competitions, hackathons, etc. regularly

Section 2: Semester-wise Structure and Curriculum for UG Course in B.Tech. Information Technology

			Semester I				
S. No	**Theory/ Practical / T&P**	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1.	Theory	IP3101	Induction Programme	-	-	-	0
2.	Theory	MA3101	Matrices and Calculus	3	1	0	4
3.	T&P	HS3101	தமிழர் மரபு / Heritage of Tamils	1	0	0	1
4.	T&P	HS3102	Communicative English – I	3	0	2	4
5.	T&P	PH3101	Engineering Physics	3	0	2	4
6.	T&P	CY3101	Engineering Chemistry	3	0	2	4
7.	T&P	CS3101	Web Development Essentials	3	0	2	4
8.	T&P	CS3102	Problem Solving Using C++ Programming	3	0	2	4
9.	Practical	ES3101	Employability Enhancement Skills – I	0	0	2	1
						Total	**26**

			Semester II				
S. No	**Theory/ Practical / T&P**	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1.	Theory	MA3201	Probability and Statistics	3	1	0	4
2.	Theory	HS3201	தமிழரும் தொழில் நுட்பமும் / Tamils and Technology	1	0	0	1
3.	T&P	HS3202	Communicative English - II	3	0	2	4
4.	T&P	EC3211	Digital System Design	3	0	2	4
5.	T&P	CS3201	Data Structures using C++	3	0	2	4
6.	T&P	CS3202	Database Management Systems	3	0	2	4
7.	T&P	CS3203	Introduction to Java Programming	3	0	2	4
8.	Practical	ES3201	Employability Enhancement Skills – II	0	0	2	1
						Total	**26**

			Semester III				
S. No	**Theory/ Practical / T&P**	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1.	Theory	MA3301	Linear Algebra	3	1	0	4
2.	Theory	IT3301	Computer Architecture and Organization	3	0	0	3
3.	T&P	IT3302	Design and Analysis of Algorithms	3	0	2	4
4.	T&P	IT3303	Web Development Practices	3	0	2	4
5.	T&P	IT3304	Computer Networks	3	0	2	4
6.	T&P	IT3305	Python Programming	3	0	2	4
7.	Practical	ES3301	Core Course Project	0	0	2	1
8.	Practical	ES3201	Employability Enhancement Skills – III	0	0	2	1
9.	MC	MC33XX	Mandatory Course – I	2	0	0	0
						Total	**25**

			Semester IV				
S. No	**Theory/ Practical / T&P**	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1.	Theory	MA3401	Discrete Mathematics	3	1	0	4
2.	Theory	IT3401	Computing Theory and Compiler Design	3	0	0	3
3.	T&P	IT3402	Operating Systems	3	0	2	4
4.	T&P	IT3403	Introduction to Artificial Intelligence and Machine Learning	3	0	2	4
5.	T&P	IT3404	Introduction to Cyber Security	3	0	2	4
6.	T&P	IT3405	Software Engineering	3	0	2	4
7.	T&P	ES3401	Core Course Project	0	0	2	1
8.	Practical	ES3201	Employability Enhancement Skills – IV	0	0	2	1
9.	MC	MC34XX	Mandatory Course – II	2	0	0	0
						Total	**25**

			Semester V				
S. No	**Theory/ Practical / T&P**	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1.	T&P	IT3501	Principles of Communication	3	0	2	4
2.	T&P	ITPE3XXX	Professional Elective-I	2/3	0	2/0	3
3.	T&P	ITPE3XXX	Professional Elective-II	2/3	0	2/0	3
4.	T&P	ITPE3XXX	Professional Elective-III	2/3	0	2/0	3
5.	T&P	IT3502	Embedded Programming	3	0	2	4
6.	T&P	IT3503	Data Analytics and Visualization	3	0	2	4
7.	Practical	ES3501	Core Course Project	0	0	2	1
8.	Practical	ES3201	Employability Enhancement Skills – V	0	0	2	1
						Total	**23**

			Semester VI				
S. No	**Theory/ Practical / T&P**	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1.	Theory	HS3601	Managerial Economics & Financial Accounting	3	0	0	3
2.	Theory	IT3601	Cloud and Distributed Computing	3	0	0	3
3.	Theory	ITOE3XXX	Open Elective- I	3	0	0	3
4.	T&P	ITPE3XXX	Professional Elective- IV	2/3	0	2/0	3
5.	T&P	ITPE3XXX	Professional Elective-V	2/3	0	2/0	3
6.	T&P	IT3602	Mobile Computing	3	0	2	4
7.	Practical	IT3603	Core Course Project	0	0	2	1
						Total	**20**

			Semester VII				
S. No	**Theory/ Practical / T&P**	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1.	Theory	HS3701	Professional Ethics and Universal Human Values	3	0	0	3
2.	Theory	ITOE3XXX	Open Elective-II	3	0	0	3
3.	T&P	ITPE3XXX	Professional Elective-VI	2/3	0	2/0	3
4.	Practical	ES3701	Internship	0	0	8	4
						Total	**13**

Semester VIII							
S. No	Theory/ Practical / T&P	Course Code	Course Title	L	T	P	C
1.	Practical	ES3801	Project Work	0	0	20	10
						Total	10

Semester I

Course Code	INDUCTION PROGRAMME	L	T	P	C
IP3101		-	-	-	-

This is a mandatory 2-week programme to be conducted as soon as the students enter the institution. Normal classes start only after the induction program is over.

The induction programme has been introduced by AICTE with the following objective:

"Engineering colleges were established to train graduates well in the branch/department of admission,have a holistic outlook, and have a desire to work for national needs and beyond. The graduating student must have knowledge and skills in the area of his/her study. However, he/she must also have broad understanding of society and relationships. Character needs to be nurtured as an essential quality by which he/she would understand and fulfill his/her responsibility as an engineer, a citizen and a human being. Besides the above, several meta-skills and underlying values are needed."

"One will have to work closely with the newly joined students in making them feel comfortable, allow them to explore their academic interests and activities, reduce competition and make them work for excellence, promote bonding within them, build relations between teachers and students, give a broaderview of life, and build character. "

Hence, the purpose of this programme is to make the students feel comfortable in their new environment, open them up, set a healthy daily routine, create bonding in the batch as well as between faculty and students, develop awareness, sensitivity and understanding of the self, people around them,society at large, and nature.

The following are the activities under the induction program in which the student would be fully engagedthroughout the day for the entire duration of the program.

(i)   Physical Activity
This would involve a daily routine of physical activity with games and sports, yoga, gardening, etc.

(ii)   Creative Arts
Every student would choose one skill related to the arts whether visual arts or performing arts. Examples are painting, sculpture, pottery, music, dance etc. The student would pursue it everyday for the durationof the program. These would allow for creative expression. It would develop a sense of aesthetics andalso enhance creativity which would, hopefully, grow into engineering design later.

(iii)   Universal Human Values
This is the anchoring activity of the Induction Programme. It gets the student to explore oneself and allows one to experience the joy of learning, stand up to peer pressure, take decisions with courage, beaware of relationships with colleagues and supporting stay in the hostel and department, be sensitive to others, etc. A module in Universal Human Values provides the base. Methodology of teaching this content is extremely important. It must not

be through do's and dont's, but get students to explore and think by engaging them in a dialogue. It is best taught through group discussions and real life activities rather than lecturing. Discussions would be conducted in small groups of about 20 students with a faculty mentor each. It would be effective that the faculty mentor assigned is also the faculty advisor for the student for the full duration of the UG programme.

(iv)   Literary Activity

Literary activity would encompass reading, writing and possibly, debating, enacting a play etc.

(v)   Proficiency Modules

This would address some lacunas that students might have, for example, English, computer familiarityetc.

(vi)   Lectures by Eminent People

Motivational lectures by eminent people from all walks of life should be arranged to give the students exposure to people who are socially active or in public life.

(vii)   Visits to Local Area

A couple of visits to the landmarks of the city, or a hospital or orphanage could be organized. This wouldfamiliarize them with the area as well as expose them to the under privileged.

(viii)   Familiarization to Dept./Branch & Innovations

They should be told about what getting into a branch or department means what role it plays in society,through its technology. They should also be shown the laboratories, workshops & other facilities.

(ix)   Department Specific Activities

About a week can be spent in introducing activities (games, quizzes, social interactions, small experiments, design thinking etc.) that are relevant to the particular branch of Engineering
/Technology/Architecture that can serve as a motivation and kindle interest in building things (become a maker) in that particular field. This can be conducted in the form of a workshop. For example, CSE and IT students may be introduced to activities that kindle computational thinking, and get them to buildsimple games. ECE students may be introduced to building simple circuits as an extension of their knowledge in Science, and so on. Students may be asked to build stuff using their knowledge of science.

Induction Programme is totally an activity-based programme and therefore there shall be no tests / assessments during this programme.

References:
Guide to Induction program from AICTE

Course Code	MATRICES AND CALCULUS	L	T	P	C
MA3101		**3**	**1**	**0**	**4**

COURSE OBJECTIVES:
1. To develop the use of matrix algebra techniques that is needed by engineers for practical applications.
2. To introduce the concept of ordinary differential equations in engineering problems.
3. To familiarize the student with the functions of several variables. This is required in many branches of engineering.
4. To acquaint the student with mathematical tools needed in evaluating multiple integrals andtheir applications.
5. To acquaint the student with the concepts of vector calculus needed for problems in all engineering disciplines.

UNIT I MATRICES   **9 + 3**

Eigenvalues and Eigenvectors of a real matrix – Characteristic equation – Properties of Eigenvalues and Eigenvectors – Cayley - Hamilton theorem – Reduction of a quadratic form to canonical form by orthogonal transformation – Nature of quadratic forms – Engineering Applications: Stretching of an elastic membrane- Capacitor- Microphone- Networks and Graphs – Communications coding theory.

UNIT II ORDINARY DIFFERENTIAL EQUATIONS   **9 +3**

Higher order linear differential equations with constant coefficients-Method of variation of parameters Homogenous equation of Euler's and Legendre's type – System of simultaneous linear differential equations with constant coefficients-Engineering Applications: Electric Circuits and Mechanical Vibrations- Simple Harmonic motion.

UNIT III FUNCTIONS OF SEVERAL VARIABLES   **9 +3**

Partial differentiation – Homogeneous functions and Euler's theorem – Total derivative – Change of variables – Jacobians –Taylor's series for functions of two variables – Engineering Applications: Maxima and minima of functions of two variables and Lagrange's method of undetermined multipliers- Transformation of co-ordinates- Functional Dependence.

UNIT IV MULTIPLE INTEGRALS   **9 +3**

Double integrals – Change of order of integration – Double integrals in polar coordinates – Area enclosed by plane curves – Triple integrals – Volume of solids – Change of variables in double and triple integrals – Engineering Applications: Moments, centres of mass, moment of inertia- Centroid – Centre of Gravity.

UNIT V VECTOR CALCULUS   **9 +3**

Gradient and directional derivative–Divergence and Curl-Vector identities–Irrotational and Solenoidal vector fields– Line integral over a plane curve–Surface integral - Area of a curved surface-Volume Integral- Green's, Gauss divergence and Stoke's theorem- Engineering Applications: Stream line in fluid dynamic-Heat transforms.

TOTAL NUMBER OF PERIODS :   60

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1	Recalling the matrix algebra methods for solving the practical problems.
CO2	Apply various techniques in solving ordinary differential equations.
CO3	Apply the differential calculus ideas on several functions.
CO4	Apply the multiple integral ideas in solving areas, volumes and other practical problems.
CO5	Analysis the Gradient, divergence and curl of a vector point function and related identities.

TEXTBOOKS:

1. Grewal B.S., "Higher Engineering Mathematics", Khanna Publishers, New Delhi, 43rd Edition, 2014.
2. Kreyszig Erwin, "Advanced Engineering Mathematics", John Wiley and Sons, 10th Edition, New Delhi, 2016.

REFERENCES:

1. Weir, M.D and Joel Hass, "Thomas Calculus", Pearson India, 12th Edition, 2016.
2. Jain R.K. and Iyengar S.R.K., "Advanced Engineering Mathematics", Narosa Publications, New Delhi, 3rdEdition, 2007.
3. O'Neil, P.V. "Advanced Engineering Mathematics", Cengage Learning India Pvt., Ltd, New Delhi, 2007.
4. Sastry, S.S, "Engineering Mathematics", Vol. I & II, PHI Learning Pvt. Ltd, 4th Edition, New Delhi, 2014.
5. Wylie, R.C. and Barrett, L.C., "Advanced Engineering Mathematics "Tata McGraw Hill   Education Pvt. Ltd, 6th Edition, NewDelhi,2012.
6. Ramana. B.V., " Higher Engineering Mathematics ", McGraw Hill Education Pvt. Ltd, New Delhi, 2016.
7. Glyn James, "Advanced Modern Engineering Mathematics", Pearson India, Eighth Edition, 2017.
8. Srimanta Pal and Subodh C. Bhunia, "Engineering Mathematics", Oxford University Press, Second Edition, 2015.

Course Code	HERITAGE OF TAMILS	L	T	P	C
HS3101		1	0	0	1

UNIT I LANGUAGE AND LITERATURE   3

Language Families in India - Dravidian Languages – Tamil as a Classical Language - Classical Literature inTamil – Secular Nature of Sangam Literature – Distributive Justice in Sangam Literature - Management Principles in Thirukkural - Tamil Epics and Impact of Buddhism & Jainism in Tamil Land - Bakthi Literature Azhwars and Nayanmars - Forms of minor Poetry - Development of Modernliterature in Tamil - Contribution of Bharathiyar and Bharathidhasan.

UNIT II HERITAGE ROCK ART PAINTINGS TO MODERN ART – SCULPTURE   3

Hero stone to modern sculpture - Bronze icons - Tribes and their handicrafts - Art of temple car making - - Massive Terracotta sculptures, Village deities, Thiruvalluvar Statue at Kanyakumari, Making of musical instruments - Mridhangam, Parai, Veenai, Yazh and Nadhaswaram - Role of Temples in Social and Economic Life of Tamils.

UNIT III FOLK AND MARTIAL ARTS   3

Therukoothu, Karagattam, Villu Pattu, Kaniyan Koothu, Oyillattam, Leather puppetry, Silambattam, Valari, Tiger dance - Sports and Games of Tamils.

UNIT IV   THINAI CONCEPT OF TAMILS   3

Flora and Fauna of Tamils & Aham and Puram Concept from Tholkappiyam and Sangam Literature - Aram Concept of Tamils - Education and Literacy during Sangam Age - Ancient Cities and Portsof Sangam Age - Export and Import during Sangam Age - Overseas Conquest of Cholas.

UNIT V CONTRIBUTION OF TAMILS TO INDIAN NATIONAL MOVEMENT AND   3
INDIAN CULTURE

                                         NUMBER OF THEORY PERIODS: 15

TEXT CUM REFERENCE
1. Social Life of Tamils (Dr. K. K. Pillay) A joint publication of TNTB & ESC and RMRL – (in print)
2. Social Life of the Tamils – The Classical Period (Dr. S. Singaravelu) (Published by: InternationalInstitute of Tamil Studies.
3. Historical Heritage of the Tamils (Dr. S. V. Subramanian, Dr. K. D. Thirunavukkarasu) (Published by:International Institute of Tamil Studies).
4. The Contributions of the Tamils to Indian Culture (Dr. M. Valarmathi) (Published by: InternationalInstitute of Tamil Studies.)
5. Keeladi – Sangam City Civilization on the banks of river Vaigai (Jointly Published by: Department of Archaeology & Tamil Nadu Text Book and Educational Services Corporation, Tamil Nadu)
6. Studies in the History of India with Special Reference to Tamil Nadu (Dr. K. K. Pillay) (Published by: The Author)
7. Porunai Civilization (Jointly Published by: Department of Archaeology & Tamil Nadu Text Book andEducational Services Corporation, Tamil Nadu)
8. Journey of Civilization Indus to Vaigai (R. Balakrishnan) (Published by: RMRL).

Course Code	COMMUNICATIVE ENGLISH –I	L	T	P	C
HS3102		3	0	2	4

COURSE OBJECTIVES:

1. To improve the communication competency.
2. To learn to use basic grammatical structures in suitable contexts.
3. To build on students' English language skills through LSRW.
4. To write in English precisely and effectively.
5. To develop language efficiently in expressing their opinions.

INTRODUCTION TO EFFECTIVE COMMUNICATION   **1**

What is effective communication? Why is communication critical for excellence during study, research, and work? What are the seven C's of effective communication?

UNIT I – INTEGRALS OF COMMUNICATION (GREETINGS &INTRODUCTION)   **11**

Listening – Listening comprehensions, listening to News
Speaking- Just A Minute, Introducing a friend
Reading- Reading Newspaper / Articles/ Magazines, Reading comprehension
Vocabulary – Synonyms& Antonyms, Acquaintance with Prefixes &suffixes from foreign languages in English toform derivatives, Word formation
Grammar – Parts of Speech, Mixed Tenses, Active &Passive Voice
Writing –Letter of Introduction, Developing the Hints

UNIT II – GIVING AND RECEIVING INSTRUCTIONS   **12**

Listening – Listen to short stories, Narrations and Persuasive speech
Speaking- Giving Instructions to use the Product/ Presenting or summarizing about a product
Reading – Reading Advertisements / Biographies
Vocabulary – Abbreviation, Acronyms, One-word SubstitutesGrammar
– WH Questions/Yes or No Questions, Imperatives Writing –
Instructions, Paragraph Writing

UNIT III – DESCRIBING PEOPLE AND PLACES   **12**

Listening- Listen to the description of a product
Speaking –Picture Description, Narrating personal experiences and events
Reading- Gadget reviews, User Manuals
Vocabulary – Homonyms, Homophones, Compound Words
Grammar – Adjectives, Adverbs, Articles.
Writing –Recommendations, Product/ Process Description.

UNIT IV –VISUALIZATION AND CLASSIFICATION   **12**

Listening – TED talks
Speaking- Interviewing a celebrity/Famous Personality
Reading – Company profiles, Business Letters Vocabulary–
Discourse Markers, Collocation, Definitions.Grammar –
Pronouns, Conjunction, Preposition
Writing – Interpretation of Charts and Graphs

UNIT V – EXPOSITION   **12**

Listening- Listen/Watch Movies / Listening to Dialogues / Conversations
Speaking- Role play, Panel Discussion, Debate Reading- Blogs, Novels, Short
Stories Vocabulary – Cause & Effect Expressions, Grammar–
Simple/Compound/Complex Sentences, Error Spotting, Punctuation.
Writing – Descriptive Essay, Dialogue Writing

List of Practical Experiments:
1. Extempore (Oral)
2. Picture Description
3. Summarizing a TED talk.
4. Conversation on asking directions, about purchasing a product.
5. Role play.

TOTAL NUMBER OF PERIODS: 60

COURSE OUTCOMES:
On completion of this course, the students will be able to:

CO1	Use appropriate words in a professional context
CO2	Gain an understanding of basic grammatical structures and use them in the right context.
CO3	Write definitions, descriptions, narrations and essays on various topics
CO4	Speak fluently and accurately in formal and informal communicative contexts
CO5	Express their opinions effectively in both oral and written medium of communication.

TEXTBOOKS:
1. English for Engineers & Technologists Orient Blackswan Private Ltd. Department of English, Anna University, (2020 edition)
2. English for Science & Technology Cambridge University Press, 2021.Authored by Dr. Veena Selvam, Dr. Sujatha Priyadarshini, Dr. Deepa Mary Francis, Dr. KN. Shoba, and Dr. Lourdes Joevani, Department of English, Anna University.

REFERENCES:
1. Technical Communication – Principles and Practices by Meenakshi Raman &Sangeeta Sharma, Oxford Univ. Press, 2016, New Delhi.
2. A Course Book on Technical English by Lakshminarayanan, Scitech Publications(India) Pvt. Ltd.

Course Code	ENGINEERING PHYSICS	L	T	P	C
PH3101		3	0	2	4

COURSE OBJECTIVES:

1. To make the students effectively understand basics of mechanics and properties of matter
2. To provide an overview of sound and ultrasonic production, detection and applications.
3. To enrich the basic knowledge of optical fiber.
4. To introduce the basics of lasers.
5. To understand quantum mechanical phenomena and apply in computing fields.

UNIT I MECHANICS AND PROPERTIES OF MATTER 9

Center of mass (CM) – motion of the CM – kinetic energy of system of particles - moment of inertia - theorems of M. I – moment of inertia of continuous bodies (Ring, Disc) – conservation of angular momentum – gyroscope. Elasticity- Hooke's law – stress –strain diagram – Factors affecting elasticity – bending of Beams -Young's modulus by uniform bending and non - uniform bending – Torsional Pendulum - I-shaped girders.

UNIT II ACOUSTICS AND ULTRASONICS 9

Acoustics: Classification and characteristics of sound-decibel- Weber-Fechner law- Factors affecting acoustics of buildings and their remedies- Methods of determination of Absorption Coefficient. Ultrasonic- Production of Ultrasonics by Magnetostriction and piezoelectric methods- acoustic grating- Non-Destructive Testing- pulse echo system through transmission and reflection modes- A, B and C- scan displays.

UNIT III LASER 9

Laser: Characteristics -Spontaneous and stimulated emission - Pumping methods- Optical Resonator – Active medium and Active centre - Einstein's coefficient - population inversion – Types of lasers - Nd-YAG laser, CO_2 laser, Semiconductor lasers: homojunction and heterojunction
Industrial and medical application

UNIT IV FIBRE OPTICS 9

Fiber optics - Principle, Numerical aperture and acceptance angle - types of optical fibers (material, refractive index, mode) - attenuation, dispersion - Fibre Optical Communication system (Block diagram) - Active and passive fiber sensors- Wavelength division multiplexing- Optical fibers in computers.

UNIT V QUANTUM MECHANICS 9

Planck's quantum theory - Compton effect - Particle properties of wave: Matter waves, wave function- The Schrodinger equation (Time dependent and time independent forms) – Particle in an infinite potential well: 1D,2D and 3D Boxes- Scanning Tunneling microscope.

NUMBER OF THEORY PERIODS: 45

List of Practical Experiments:

1. Ultrasonic interferometer – determination of the velocity of sound and compressibility of liquids
2. Simple harmonic oscillations of cantilever.
3. Uniform bending – Determination of Young's modulus.
4. Laser- Determination of the wave length of the laser using grating.
5. Optical fibre -Determination of Numerical Aperture and acceptance angle.
6. Acoustic grating- Determination of velocity of ultrasonic waves in liquids.
7. Michelson Interferometer

NUMBER OF PRACTICAL PERIODS: 15
TOTAL NUMBER OF PERIODS : 16

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1	Understand the importance of mechanics and express their knowledge in properties of matter
CO2	Analyze the applications of acoustics and ultrasonic in engineering field.
CO3	Demonstrate a strong foundational knowledge in fiber optics
CO4	Acquire knowledge in laser and its applications.
CO5	Comprehend and apply quantum mechanical principles.

TEXTBOOKS:

1. D. Kleppner and R. Kolenkow. An Introduction to Mechanics. McGraw Hill Education(Indian Edition), 2017.
2. Arthur Beiser, Shobhit Mahajan, S. Rai Choudhury, Concepts of Modern Physics,McGraw-Hill (Indian Edition), 2017.
3. Gaur R.K. and Gupta S.L. Engineering Physics. Dhanpat Rai publishers, 2009.
4. Kasap, Safa, Capper, "Handbook of Electronic and Photonic Materials" - 2nd edition, Springer, 2017.
5. Eleanor Rleffel and Wolfgang Polak, "Quantum computing a gentle introduction", 1st edition, The MIT press,2012.

REFERENCES:

1. R.Wolfson. Essential University Physics. Volume 1 & 2. Pearson Education (IndianEdition), 2009.
2. K.Thyagarajan and A.Ghatak. Lasers: Fundamentals and Applications, Laxmi Publications, (Indian Edition), 2019.
3. D.Halliday, R.Resnick and J.Walker. Principles of Physics, Wiley (Indian Edition),2015.
4. Searls and Zemansky. University Physics, 2009.
5. David J. Griffiths, "Introduction to Quantum Mechanics", 2nd edition, Cambridgeuniversity press, 2017.
6. Chris Bernhardt, "Quantum computing for everyone" The MIT press, 2019.

Course Code	ENGINEERING CHEMISTRY	L	T	P	C
CY3101		3	0	2	4

COURSE OBJECTIVES:
1. To inculcate sound understanding of water quality and water treatment techniques.
2. To impart knowledge on the preparatory methods of nanomaterials. To introducethe properties and applications of composites
3. To facilitate the understanding of fuel classification, preparation, combustion, andenvironmental impact.
4. To conversant with the principle of chemistry, cell reactions, and corrosionprotection techniques.
5. To acquire a deep understanding of renewable energy sources along with energystorage technologies and innovation in sustainable energy systems.

9

UNIT I WATER AND ITS TREATMENT
Water: Sources and impurities, hardness, alkalinity. Treatments - sterilization - break point chlorination, UV, Ozonation, Desalination of brackish water: Reverse Osmosis. Boiler troubles: Scale and sludge, Boiler corrosion, Caustic embrittlement, Priming & foaming. Treatment of boiler feed water: Internal treatment (phosphate, colloidal, sodium aluminate and calgon conditioning) and External treatment Ion exchange demineralization and zeoliteprocess.

9

UNIT II NANO MATERIALS & COMPOSITE
Basics: Distinction between molecules, nanomaterials and bulk materials; Types of nanomaterials: Definition, properties and uses of nano particles and nanotube. Preparationof nano materials: laser ablation, and electro spinning. An application of nano materials in medicine, agriculture, energy, electronics and catalysis.
Composite: Properties and applications of: Metal matrix composites (MMC), Ceramic matrix composites and Polymer matrix composites. FRP- Hybrid composites - definition andexamples.

UNIT III FUELS AND COMBUSTION   **9**
Fuels: Introduction: Classification of fuels; Coal and coke: Analysis of coal (proximate and ultimate), Carbonization, Manufacture of metallurgical coke (Otto Hoffmann method). Petroleum and Diesel: Manufacture of synthetic petrol (Bergius process), Knocking - octane number, diesel oil - cetane number; Power alcohol and biodiesel.
Combustion of fuels: Introduction: Calorific value - higher and lower calorific values, Theoretical calculation of calorific value; Flue gas analysis - ORSAT Method. CO_2 emissionand carbon foot print.

UNIT IV ELECTRO CHEMISTRY AND CORROSION CONTROL   **9**
Electrochemistry - Introduction, Electrochemical cells - electrolytic cell - reversible and irreversible cells. Electrode potential - Oxidation and reduction Potentials - emf, Nernst equation and applications. Reference electrodes - Calomel electrode - Electrochemical series - its applications.
Corrosion protection - cathodic protection - sacrificial anodic and impressed current protection methods; advanced protective coatings: electroplating and electroless plating.

UNIT V ENERGY SOURCES AND STORAGE DEVICES   **9**

Solar energy conversion: Principle, working and applications of solar cells; recent developments in solar cell materials. Wind energy; Geothermal energy. Hydrogen as fuel - Sources of hydrogen - Hydrogen production methods – electrolysis, limitations and applications. Storage Devices: Batteries - Types of batteries, Primary battery - dry cell, Secondary battery n- lead acid battery and lithium-ion-battery; Electric vehicles-working principles; Fuel cells:H2-O2 fuel cell.

NUMBER OF THEORY PERIODS: 45

List of Practical Experiments:
1. Determination of total, temporary and permanent hardness of water by EDTA method.
2. Determination of alkalinity in water sample.
3. Estimation of chloride content in water by Argentometric method [Mohr's Method].
4. pH-metry determination strength of HCl by NaOH.
5. Determination of $BaCl_2$ by conductometric precipitation titration.
6. Conductometric titration of mixture of acids (HCl &CH_3COOH).
7. Estimation of iron content of the given solution by using potentiometer.

NUMBER OF PRACTICAL PERIODS : 15
TOTAL NUMBER OF PERIODS   **:** 16

COURSE OUTCOMES:
On completion of this course, the students will be able to:

CO1	Analyse the quality of water and propose suitable treatment methodologies for hard water.
CO2	Identify and apply basic concepts of nanomaterial's preparation for engineering applications.
CO3	Acquire knowledge of fuel properties, manufacturing processes, combustion characteristics, and environmental considerations.
CO4	Demonstrate chemical principles, cell reactions and corrosionprotection techniques.
CO5	Analyse proficiency in different forms of energy resources and fuel cell utilization, fostering the lead advancements in renewable energy and energy storage solutions.

TEXTBOOKS:
1. Dara S.S, Umare S.S, "Engineering Chemistry", First revised Edition by S. Chand & Company Ltd., New Delhi 2015.
2. Jain P. C. & Monica Jain., "Engineering Chemistry", 16th Edition, Dhanpat RaiPublishing Company (P) Ltd, New Delhi, 2015.
3. K. Klabunde, G. Sergeev, Nanochemistry, 2nd Edition, Springer Publisher, 2013.
4. S.A. Sherif, D. Yogi Goswami, E.K. (Lee) Stefanakos, Aldo Steinfeld "Handbook of Hydrogen Energy", 1st Edition, CRC Press, 2014.

REFERENCES:

1. Shikha Agarwal., "Engineering Chemistry and Applications", Cambridge University press, 2016.
2. Lefrou.,Christine., Fabry., Pierre., Poignet., Jean-claude., "Electrochemistry - The Basics, with examples" Springer. 2012.
3. Zaki Ahmad, Digby Macdonald, "Principles of Corrosion Engineering and Corrosion Control", Elsevier Science, 2nd Edition, 2012.
4. Introduction to Nano: basics to Nanoscience and Nanotechnology, by Sengupta, Amretashis, Sarkar, Chandan Kumar, Springer Publisher, 2015.
5. Kazunari Sasaki, Hai-Wen Li, Akari Hayashi, Junichiro Yamabe, Teppei Ogura, Stephen M. Lyth, Hydrogen Energy Engineering A Japanese Perspective, Springer, 2016

Course Code	WEB DEVELOPMENT ESSENTIALS	L	T	P	C
CS3101		3	0	2	4

COURSE OBJECTIVES:

1. To understand the web programming using Hypertext Markup Language (HTML)
2. To develop responsive websites using Cascading Style Sheets (CSS)
3. To implement interactive and dynamic effects on web pages using JavaScript.
4. To develop dynamic web applications using Document Object Model (DOM).
5. To develop dynamic web applications with Asynchronous JavaScript.

UNIT I - HYPERTEXT MARKUP LANGUAGE (HTML)   **12**
(Blooms Learning Levels: L3 – Apply)

Introduction to HTML – HTML Document Structure –Text – Lists -Frames, Table and Divtags – Forms -Images –Hyperlinks - Tag Attributes – Structuring Webpage - Semantic HTML.

Coding Exercises:
   1. Design a web page using HTML basic tags (text elements only).
   2. Develop web site with suitable contents and links (use text, form and anchor tag).
   3. Design web pages using lists and table tags.

Problems-solving Assignments:
   1. Develop a web application for commercial organization like restaurant, spa services etc., using HTML tags (Text, Lists, Frames, Table, Div, Images, Hyperlinks tags)

UNIT II – CASCADING STYLESHEET (CSS)   **12**
(Blooms Learning Levels: L3 – Apply)

Introduction to CSS - Inline, Internal and External CSS - Styling Text - Combining Selectors - Class and ID Selectors - Working with Colors - Styling Hyperlinks – Layouts: Float, Flexbox and CSS Grid – Web Design Rules and Frameworks.

Coding Exercises:
   1. Design a web page using using HTML and CSS.
   2. Develop a web site with suitable CSS layouts.

Problems-solving Assignments:
1. Develop a web application for higher educational institutions – college/university using HTML and CSS (apply inline, internal and external styles - Use style propertiesfor background, text effects, positioning, link).

UNIT III – JAVASCRIPT (Blooms Learning Levels: L3 – Apply   **12**

Introduction to JavaScript – Variables - Data Types – Operators - Statements and Expressions- Strings and Template Literals - if / else Statements - switch Statement - Functions – Arrays – Classes and Objects-Looping Statements.

Coding Exercises:
1. Write a javascript program to Check if a Number is Odd or Even.
2. Write a javascript program to find sum of natural numbers.

Problems-solving Assignments:

1. Develop a web page for creating online bank account with Login, Registration form, and Dashboard with drop down   menus. Perform validation on the form elements data such as entering valid data for required fields or not.

UNIT IV – DOCUMENT OBJECT MODEL (DOM)(Blooms Learning Levels: L3 –Apply)   12

Introduction to DOM – HTML elements access using DOM (find, change the content and replace/remove HTML elements) - Events Fundamentals – event listener and its methods– mouse, keyboard and from events

Coding Exercises:
1. Write a javascript program to find and change the content of the HTML elements.
2. Write a javascript program to handle events (Mouse/Keyboard/Form events).

Problems-solving Assignments:
1. Write a javascript program to perform replacement and removal of HTML elements.

UNIT V – ASYNCHRONOUS JAVASCRIPT ~~AND GITHUB~~ (Blooms Learning   12
Levels: L3 – Apply)

JavaScript: Behind the scene - JS Callbacks - JS Asynchronous - JS Promises - JS Async/Await

Coding Exercises:

1. Write a javascript program to display a text using setTimeout method.
2. Write a program to perform javascript callback function.

Problems-solving Assignments:

1. Write a javascript program to demonstrate the use of javascript promise.

Final Project: Design and develop a web application for an E-Commerce portal (like Flipkart, Amazon) with the following features:
1. Create a menu item to navigate entire website (Home, Product items, Contact Us, About Us)
2. Create and check the input field validation for User Registration and Login form
3. Display the items of Orders list, Wishlist, comments & reviews, Customer Care and reporting options on the webpage.
4. Uploading the Project in the GitHub Repository.

TOTAL NUMBER OF PERIODS : 60

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Implement websites using HTML elements
CO2   Apply styling to HTML content using CSS
CO3   Implement dynamic effects using JavaScript
CO4   Manipulate the webpage elements using DOM
CO5   Develop webpage with asynchronous javascript

TEXTBOOKS:

1. Jennifer Niederst Robbins, "Learning Web Design, A beginner's guide to HTML, CSS,JavaScript, and Web Graphics", O'Reilly Media, 5th Edition, 2018.

2. Ben Frain, "Responsive Web Design with HTML5 and CSS: Build future-proof ResponsiveWebsites using latest HTML5 and CSS techniques", 4th Edition, 2023.

3. David Flanagan, "JavaScript: The Definitive Guide: Master the World's Most-Used Programming   Language", 7th Edition, O'Reilly Media, Inc.,2020.

4. "HTML and CSS QuickStart Guide: The Simplified Beginners Guide to Developing a Strong Coding Foundation, Building Responsive Websites, and Mastering the Fundamentals of Modern Web Design" by David DuRocher (2021)

5. "JavaScript: The Definitive Guide" by David Flanagan (7th Edition, 2020)

REFERENCES:

1. P.J. Deitel, H.M. Deitel, and Nieto, "Internet and World Wide Web - How to Program", Prentice Hall, 6th Edition, 2020.

2. "Learning JavaScript: Add Sparkle and Life to Your Web Pages" by Ethan Brown (3rd Edition, 2021)

3. "JavaScript Everywhere: Building Cross-Platform Applications with GraphQL, React, React Native, and Electron" by Adam D. Scott (2020)

Course Code	PROBLEM-SOLVING USING C++ PROGRAMMING	L	T	P	C
CS3102		3	0	2	4

COURSE OBJECTIVES:

1. To learn the concepts of object-oriented programming using C++

2. To explore function overloading and constructors in C++

3. To gain knowledge of the concepts inheritance and polymorphism in C++

4. To gain comprehension about the concepts of exception handling in C++

5. To learn the concepts Files in C++

UNIT I OVERVIEW OF C   **12**

Introduction to C- C Program Structure- Basic Syntax and Structure, Variables and Data Types, Operators, Decision control and loops, Arrays, String, Functions and Structures.

Coding Exercises:

1. Implement a C program to print half pyramid using '*'.
2. Write a C program to find the GCD (Greatest Common Divisor) of two numbers using a function.
3. Write a C program to check if a given number is a prime number.

Problems-solving Assignments:

1. Write a C program to calculate the factorial of a given positive integer using a loop.
2. Implement electricity bill using C.

UNIT II   FUNCTION OVERLOADING AND CONSTRUCTORS   12
(Blooms Learning Levels: L3 – Apply)

Function Overloading – Overloading Constructors – Copy Constructors – Finding the address of overloaded Function– Default Function Arguments– Function Overloading and Ambiguity.

Coding Exercises:

1. Implement a C++ program to create a class called "simple class". Create a constructor and destructor for this class called simple class.
2. Implement a C++ program for a Copy Constructor. Create a Person class with a name and an age and create a copy constructor to create a new object with the same name and age as the source object.
3. Implement a C++ program for Overloading Functions with Different Number of Parameters for addition in a calculator.

Problems-solving Assignments:

1. Write a C++ program to sort an array of strings in alphabetical order Implement a recursive function to calculate the factorial of a given number using c++
2. Write a C++ program to reverse a linked list.

UNIT III INHERITANCE AND POLYMORPHISM   12
(Blooms Learning Levels:L3 – Apply)

Inheritance: Base-Class Access Control – Inheritance and Protected Members – Inheriting Multiple Base Classes – Constructors, Destructors and Inheritance – Granting Access – Virtual Base Classes. Virtual functions and Polymorphism: Virtual Functions – Virtual Attribute and Inheritance – Virtual Functions and Hierarchy – Pure Virtual Functions – Using Virtual Functions – Early vs. Late Binding. Run-Time Type ID and Casting Operators: RTTI
Casting Operators – Dynamic Cast.

Coding Exercises:
1. Implement a C++ program for Inheritance for calculating the area of a triangle.
2. Implement a C++ program that demonstrates polymorphism using a basic example of shapes.

Problems-solving Assignments:

1. Design a class hierarchy for a university. Create a base class Person with attributes like name, age, and gender. Derive classes Student and Professor from the base class. Implement methods to display information about each person type. Use polymorphism to create an array of Person pointers containing both students and professors and display their information.

2. Create a base class called Employee with attributes like name and salary. Derive two classes, Manager and Worker, from the base class. The Manager class should have an additional attribute for bonus, while the Worker class should have an attribute for hours worked and an hourly wage. Implement virtual methods for calculating the total earnings (salary + bonus for managers, hourly wage * hours worked for workers). Create instances of managers and workers and display their total earnings.

UNIT IV   TEMPLATES AND EXCEPTION HANDLING   12
(Blooms Learning Levels: L3 – Apply)

Templates: Generic Functions – Applying Generic Functions – Generic Classes – Type name and Export Keywords – Power of Templates. Exception Handling: Fundamentals – Handling Derived Class Exceptions – Exception Handling Options – Understanding terminate() and unexpected() – uncaught exception() Function – Exception and bad exception Classes – Applying Exception Handling.

Coding Exercises:

1. Class Templates – Define a template for example stack. Define a template parameter type name T which will represent the data type that the stack will hold. Define the class methods push, pop, empty and size having their respective data types. Create two instance of the stack in the main function, one for integers and the other for double. Perform stack operations.

2. Create a max function template using template keyword. Create a template parameter using the declaration <typename T>. this template should act as place holder for the actual type that will be used when the function is instantiated. The max function created takes two parameters of the type T and should return the maximum of two values. Create a main function. This template should be used by calling both integer and double values.

3. In this program, the try block contains the code that might potentially raise an exception. In this case, it attempts to perform a division operation and throws a std::runtime_error exception if the denominator is zero. The catch block catches any exception of type std::exception (or its derived classes) and displays an error message.

4. Using c++ programming perform division based on user inputs for numerator and denominator. If the user enters a denominator of 0, a std::runtime_error exception must be thrown with a custom error message. The try block must contain the code that might throw an exception, and the catch block must catch the exception and display the error message using the what() function of the exception object. Regardless of whether an exception is thrown or not, the program must continue executing after the exception handling block.

Problems-solving Assignments:

1. Define a custom exception class called "Negative Number Exception." Write a program that takes an integer as input. If the input is negative, throw an instance of this custom exception. Use a try-catch block to catch and handle the custom exception, displaying an error message.

2. Create a program that performs a division of two integers, but this time within a loop that allows the user to keep trying until they provide valid input. Use nested try-catch blocks to handle exceptions at different levels of the program's execution.

UNIT V   I/O STREAMS   (Blooms Learning Levels: L3 – Apply)   12

File I/O-fstream and the File Classes-Opening and Closing a File-Reading and Writing Text Files-Unformatted and Binary I/O. Namespaces: Namespaces – std namespace.

Coding Exercises:

1. Implement a c++ program which includes the necessary header files: <iostream> for input/output operations and <fstream> for file stream operations. Use an ofstream object to write data to the file named "output.txt". There should be a check if the file is opened successfully. Use << operator to write data to the file, and use close() method for closing the file. Use ifstream object to read data from the same file, use close() method for closing the file. The program must return 0 to indicate successful execution.

Problems-solving Assignments:

1. Write a C++ program to read student records from a file and calculate their total and percentage
2. Develop a program to read employee details from a file, sort them based on salary, and write the sorted data back to the file Final Project:
3. Design a program to manage a library's inventory, allowing users to add, remove, and search for books, while also keeping track of borrowed and returned books.
4. Develop a system that simulates basic bank account operations like deposits, withdrawals, and balance inquiries. You could use classes to model accounts.
5. Design a simple inventory management system for a small store using structs and file handling. The program should allow users to add, update, and delete items in the inventory and display the current stock

TOTAL NUMBER OF PERIODS : 60

COURSE OUTCOMES:
On completion of this course, the students will be able to:

CO1	Implement basic concepts of structures, union and pointers using C++
CO2	Implement function overloading and constructors using C ++
CO3	Implement inheritance and polymorphism using C ++
CO4	Implement templates and exception handling using C ++
CO5	Implement I/O streams using C ++ and develop simple applications.

TEXTBOOKS:
1. Herbert Schildt, "C++: The Complete Reference", 5th Edition, Tata Mc-Graw Hill Publishers, 2014.
2. Paul Deitel, Harvey Deitel, "C++ How to Program", 8th Edition, Prentice Hall Publisher, 2016.
3. Trivedi, Bhushan "Programming with ANSI C++", 2nd Edition, Oxford UniversityPress NASW Press, 2013.

REFERENCES:
1. Ira Pohl, "Object Oriented Programming using C++", 2nd Edition, Pearson Education,Reprint, 2004.
2. S. B. Lippman, Josee Lajoie, Barbara E. Moo, "C++ Primer", 4th Edition, PearsonEducation, 2012.
3. Bjarne Stroustrup, "The C++ Programming language", 4th Edition, Pearson Education, 2013.

Course Code	EMPLOYABILITY ENHANCEMENT SKILLS – I FOUNDATIONS OF MATHEMATICS AND REASONING	L	T	P	C
ES3101		0	0	2	1

COURSE OBJECTIVES:

The main objectives of this course are to:

- To categorize, apply and use thought process to understand the concepts of Quantitative methods to enhance problem solving skills.
- To prepare and explain the fundamentals related to various possibilities with numeric ability and probabilities related to quantitative aptitude.
- To critically evaluate numerous possibilities related to puzzles.

UNIT I NUMBERS 9

Introduction – Classification of numbers – Formation of Numbers (Small & Large) –Place Value – Face Value – Divisibility Rule – Prime, Composite Numbers – Prime Factorization – Number of factors – Number of factors (Odd & Even) – Sum of factors – Successors and Predecessors – Greatest Integer Value – Vedic Mathematics – Trailing Zeroes – Unit Digits–Remainder Theorem – Real Number – Rational Numbers: Integers, Fractions – Comparison of Numbers – Operations on fractions – Scientific Notation.

UNIT II PROBLEMS ON LETTERS, NUMBERS AND SYMBOLS 9

Factors and Multiples, LCM and HCF – Relationship between LCM and HCF – Factorial – Simplification – VBODMAS – Square, Square Root – Cube, Cube Root – Exponents & Powers (Surds and Indices) – Sequence & Series: Arithmetic Progression – Geometric Progression – Special Progression, Letter Series, Number Series, Alpha – Numeric Series, Continuous Pattern Series

UNIT III VERBAL AND NON-VERBAL REASONING 9

Verbal Reasoning – Analogy: Completing the Analogous pair, Direct Analogy, Choosing the Analogous pair, Double Analogy, Choosing a Similar Word, Detecting Analogies, Multiple word Analogy, Number Analogy, Alphabet Analogy – Classification: Odd Words and Numerals – Coding and Decoding: Letter, Number, Symbol, Matrix, Substitution, Deciphering Message Word, Number and Symbols. Non – Verbal Reasoning Figure Series – Missing figure, Incorrect figure – Analogy: Similarity Related Pair, Similarity Related figures, unrelated figures, Group of figures.

UNIT IV RATIO AND PROPORTION 9

Introduction – Ratio – Proportion: Direct and Indirect – Unitary Method – Problems on Ages – Chain Rule – Partnership – Mixture or Allegation – Time and Work: Individual, Group, Efficiency, Wages – Pipes and Cistern: Inlet, Outlet, and Leakage

UNIT V PERCENTAGE 9

Introduction – Percentages in real life – Profit and Loss – Discount– Simple Interest – Compound Interest – Relationship between Simple Interest and Compound Interest – Overhead Expenses and GST.

COURSE OUTCOMES:

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

CO1	Develop the arithmetic ability and properties of numbers that we use in day-to-day life.
CO2	Demonstrate the logic behind the formation of numbers, alphabets series.
CO3	Apply the reasoning methods logically and evaluate complex relationships between the variables and numbers
CO4	Use the concept of ratios and proportion in ages and partnership problems.
CO5	Apply the short cuts of the mathematical tricks to reduce the time duration in problem solving.

TEXT BOOKS:

1. Aggarwal, R.S, "Quantitative Aptitude for Competitive Examinations (Revised edition)", S Chand Publishing, 2017.
2. Arun Sharma, "Teach Yourself Quantitative Aptitude", McGraw Hill Education, 2017
3. Aggarwal, R.S, "A Modern Approach to Verbal & Non-Verbal Reasoning" 2nd edition, S Chand Publishing, 2018.

REFERENCES:

1. Akhilesh Khare, "Shortcuts in Mathematics", CreateSpace Independent Pub, 2015.
2. Ravi Shankar, "Vedic Maths for Competitive Exams", Pustak Mahal, 2016.
3. Abhijit Guha, "Quantitative Aptitude for Competitive Examinations", 7th edition, McGraw Hill Education, 2020.

TOTAL: 30 PERIODS

Course Code		L	T	P	C
MA3201	**PROBABILITY AND STATISTICS**	**3**	**1**	**0**	**4**

COURSE OBJECTIVES:
1. To introduce the basic concepts of probability and random variables
2. To introduce the basic concepts of two-dimensional random variables
3. To acquire the knowledge of testing of hypothesis for small and large samples this plays an important role in real life problems.
4. To introduce the basic concepts of classifications of design of experiments which plays very important roles in the field of agriculture.
5. To apply the statistical tools in engineering problems and quality control.

UNIT I - ONE DIMENSIONAL RANDOM VARIABLES   **12**

Probability –Discrete and continuous random variables – Moments – Moment generating functions – Binomial, Poisson, Geometric, Uniform, Exponential and Normal distributions.

UNIT II - TWO DIMENSIONAL RANDOM VARIABLES   **12**

Joint distributions – Marginal and conditional distributions – Covariance – Correlation and linear regression – Transformation of random variables – Central limit theorem (excluding proof).

UNIT III - TESTING OF HYPOTHESIS   **12**

Sampling distributions – Estimation of parameters – Statistical hypothesis – Large sample tests
based on Normal distribution for single mean and difference of means – Tests based on t, Chi - square and F distributions for mean, variance – Contingency table (test for independent) – Goodness of fit.

UNIT IV - DESIGN OF EXPERIMENTS   **12**

One way and two-way classifications – Completely randomized design – Randomized block design – Latin square design

UNIT V - STATISTICAL QUALITY CONTROL   **12**

Control charts for measurements (X and R charts) – Control charts for attributes (p, c and np charts) – Tolerance limits – Acceptance sampling.

TOTAL NUMBER OF PERIODS: 60

COURSE OUTCOMES:
On completion of this course, the students will be able to:

CO1	Understand the fundamental knowledge of the concepts of probability and have knowledge of standard distributions which can describe real life phenomenon
CO2	Understand the basic concepts of two-dimensional random variables and apply in engineering applications
CO3	Apply the concept of testing of hypothesis for small and large samples in real life problems
CO4	Apply the basic concepts of classifications of design of experiments in the field of agriculture.
CO5	Develop skills to analyze quality related data using advanced statistical methods.

TEXTBOOKS:
1. Johnson, R.A., Miller, I and Freund J., "Miller and Freund's Probability and Statistics for Engineers", Pearson Education, Asia, 8th Edition, 2015.

2. Milton. J. S. and Arnold. J.C., "Introduction to Probability and Statistics", Tata McGraw Hill, 4th Edition, 2007.

3. Walpole. R.E., Myers. R.H., Myers. S.L. and Ye. K., "Probability and Statistics for Engineers and Scientists", Pearson Education, Asia, 8th Edition, 2007.

REFERENCES:
1. Devore. J.L., "Probability and Statistics for Engineering and the Sciences", Cengage Learning, New   Delhi, 8th Edition, 2014
2. Papoulis, A. and Unni krishnapillai, S., "Probability, Random Variables and Stochastic Processes",McGraw   Hill Education India, 4th Edition, New Delhi, 2010.
3. Ross, S.M., "Introduction to Probability and Statistics for Engineers and Scientists", 3rd Edition,   Elsevier, 2004.
4. Spiegel. M.R., Schiller. J. and Srinivasan, R.A., "Schaum's Outline of Theory and Problems of   Probability and Statistics", Tata McGraw Hill Edition, 2004.
5. 5. Gupta S.C. and Kapoor V.K., "Fundamentals of Mathematical Statistics", Sultan Chand & Sons, New Delhi, 12th Edition, 2020.

Course Code	TAMILS AND TECHNOLOGY	L	T	P	C
HS3201		1	0	0	1

3

UNIT I WEAVING AND CERAMIC TECHNOLOGY
Weaving Industry during Sangam Age – Ceramic technology – Black and Red Ware Potteries
(BRW) – Graffiti on Potteries

3

UNIT II DESIGN AND CONSTRUCTION TECHNOLOGY
Designing and Structural construction House & Designs in household materials during
Sangam Age – Building materials and Hero stones of Sangam age – Details of Stage
Constructions in Silappathikaram – Sculptures and Temples of Mamallapuram – Great
Temples of Cholas and other worship places – Temples of Nayaka Period – Type study
(Madurai Meenakshi Temple) – Thirumalai Nayakar Mahal – Chetti Nadu Houses, Indo –
Saracenic architecture at Madras during
British Period.

UNIT III MANUFACTURING TECHNOLOGY   3

Art of Ship Building – Metallurgical studies – Iron industry – Iron smelting,steel – Copper
and goldCoins as source of history – Minting of Coins – Beads making – industries Stone beads
– Glass

beads – Terracotta beads – Shell beads/ bone beats – Archeological evidences – Gem stone
types described in Silappathikaram.

UNIT IV AGRICULTURE AND IRRIGATION TECHNOLOGY   3

Dam, Tank, ponds, Sluice, Significance of Kumizhi Thoompu of Chola Period, Animal
Husbandry
– Wells designed for cattle use – Agriculture and Agro Processing – Knowledge of Sea –
Fisheries – Pearl – Conche diving – Ancient Knowledge of Ocean – Knowledge Specific Society.

UNIT V SCIENTIFIC TAMIL & TAMIL COMPUTING   3

Development of Scientific Tamil – Tamil computing – Digitalization of Tamil Books –
Development
of Tamil Software – Tamil Virtual Academy – Tamil Digital Library – Online Tamil Dictionaries
– Sorkuvai Project.

TOTAL NUMBER OF PERIODS :   15

TEXT – CUM – REFERENCE BOOKS:

1. **Social Life of Tamils (Dr.K.K.Pillay) A joint publication of TNTB & ESC and RMRL – (in print)**
2. **Social Life of the Tamils – The Classical Period (Dr.S.Singaravelu) (Published by: International Institute of Tamil Studies.**
3. **Historical Heritage of the Tamils (Dr.S.V.Subatamanian, Dr.K.D. Thirunavukkarasu) (Published by: International Institute of Tamil Studies).**
4. **The Contributions of the Tamils to Indian Culture (Dr.M.Valarmathi) (Published by: International Institute of Tamil Studies.)**
5. **Keeladi – 'Sangam City C ivilization on the banks of river Vaigai' (Jointly Published by: Department of Archaeology & Tamil Nadu Text Book and Educational Service Corporation Tamil Nadu)**
6. **Studies in the History of India with Special Reference to Tamil Nadu (Dr.K.K.Pillay) (Published by: The Author)**
7. **Porunai Civilization (Jointly Published by: Department of Archaeology & Tamil**

Nadu Text Book and Educational Services Corporation, Tamil Nadu)

8. **Journey of Civilization Indus to Vaigai (R.Balakrishnan) (Published by: RMRL) – Reference Book.**

Course Code	COMMUNICATIVE ENGLISH –II	L	T	P	C
HS3202		3	0	2	4

COURSE OBJECTIVES:
1. To engage learners in meaningful language activities to improve their LSRW skills
2. To enhance learners awareness of general rules of writing for specific audiences
3. To help learners understand the purpose, audience, contexts of different types of writing
4. To develop analytical thinking skills for problem solving in communicative contexts
5. To cultivate the learners to understand job applications and interviews for internship and placements.

9

UNIT I - MAKING COMPARISONS
Listening – Evaluative Listening: Advertisements, Product Descriptions, – Audio/ video Speaking– Marketing a product, Persuasive Speech Techniques. Reading – Reading advertisements, user manuals, brochures Writing – Letter to the editor; Compare and Contrast Essay Grammar –
Impersonal passive voice; Prepositional phrases Vocabulary –Contextual meaning of words.

9

UNIT II - EXPRESSING CASUAL RELATIONS IN SPEAKING AND WRITING
Listening – Listening to longer technical talks and completing–gap filling exercises. Listening technical information from podcasts Speaking –Describing and discussing the reasons of accidents or disasters based on news reports Reading – Reading longer technical texts/Novels Writing – Writing responses to complaints; Problem solution Essay Grammar –Subject – Verb Agreement,
Infinitive and Gerunds Vocabulary – Adverbs.

9

UNIT III - PROBLEM SOLVING
Listening–Watching movie scenes/documentaries depicting a technical problem and suggesting solutions. Speaking – Group Discussion (based on case studies), – techniques and Strategies. Reading – Case Studies, excerpts from literary texts, news reports etc. Writing – Checklists, Argumentative Essay Grammar –Error correction; If conditional sentences Vocabulary – Compound
Words, Sentence Completion.

9

UNIT IV - REPORTING OF EVENTS AND RESEARCH
Listening – Listening Comprehension based on news reports and documentaries. Speaking – Interviewing, Presenting an oral report, Mini presentations on select topics Reading – Newspaper articles Writing –Industrial visit Report, Accident Report, Survey Report Grammar–Reported
Speech, Modals Vocabulary–Conjunctions, use of prepositions.

9

UNIT V - THE ABILITY TO PUT IDEAS OR INFORMATION COGENTLY
Listening – Listening to TED Talks, Presentations, Formal job interviews Speaking – Mock Interview, Making presentations with visual aids Reading – Company profiles, Statement of Purpose, (SOP), an excerpt of interview with professionals Writing –Job / Internship application – Cover letter
with Resume Grammar – Numerical adjectives, Relative Clauses Vocabulary–Idioms.

NUMBER OF THEORY PERIODS :   45

List of Practical Experiments :
1. Listening /Reading Comprehension
2. Introducing Oneself
3. Summary of a Podcast
4. Mini Presentation on General topic (ICT tools)
5. Autobiography of a famous Personality
6. Narrating an unforgettable event
7. Drafting an Email (printed format)
8. Developing a story using given Vocabulary
9. Group Discussion
10. Mock Interview

NUMBER OF PRACTICAL PERIODS :	30
TOTAL NUMBER OF PERIODS :	75

COURSE OUTCOMES:
On completion of this course, the students will be able to:
CO1 Compare and contrast products and ideas in technical texts.
CO2 Identify cause and effects in events, industrial processes through technical texts
CO3 Analyse problems in order to arrive at feasible solutions and communicate them orally and in the written format.
CO4 Report events and the processes of technical and industrial nature
CO5 Present their opinions in a planned and logical manner, and draft effective resumes in context of job search.

TEXTBOOKS:
1. English for Engineers & Technologists (2020edition) Orient Blacks wan Private Ltd Department of English, Anna University.
2. English for Science & Technology Cambridge University Press 2021. Authored by Dr. Veena Selvam, Dr. Sujatha Priyadarshini, Dr. Deepa Mary Francis, Dr. KN. Shoba, and Dr. Lourdes Joevani, Department of English, Anna University.
3. Professional English-II, V.K.Publications, Dr.S.N. Mahalakshmi.

REFERENCES:
1. Raman. Meenakshi, Sharma. Sangeeta (2019). Professional English. Oxford University Press. New Delhi.
2. Improve Your Writing. V.N. Arora and Laxmi Chandra, Oxford Univ. Press, 2001, New Delhi.
3. Learning to Communicate–Dr.V. Chellammal. Allied Publishers, New Delhi,2003
4. Business Correspondence and Report Writing by Prof. R.C. Sharma & Krishna Mohan, Tata McGraw Hill & Co. Ltd., 2001, New Delhi.

Course Code	DIGITAL SYSTEM DESIGN	L	T	P	C
EC3211		3	0	2	4

COURSE OBJECTIVES:

- To present the fundamentals of digital circuits and simplification methods
- To practice the design of various combinational digital circuits using logic gates
- To bring out the analysis and design procedures for synchronous and asynchronous Sequential circuits
- To learn integrated circuit families.
- To introduce semiconductor memories and related technology

UNIT I BASIC CONCEPTS   **9**

Review of number systems-representation-conversions, Review of Boolean algebra-theorems, sum of product and product of sum simplification, canonical forms min term and max term, Simplification of Boolean expressions-Karnaugh map, completely and incompletely specified functions, Implementation of Boolean expressions using universal gates, Tabulation methods.

UNIT II COMBINATIONAL LOGIC CIRCUITS   **9**

Problem formulation and design of combinational circuits - Code-Converters, Half and Full Adders, Binary Parallel Adder – Carry look ahead Adder, BCD Adder, Magnitude Comparator, Decoder, Encoder, Priority Encoder, Mux/Demux, Case study: Digital trans- receiver / 8 bit Arithmetic and logic unit, Parity Generator/Checker, Seven Segment display decoder

UNIT III SYNCHRONOUS SEQUENTIAL CIRCUITS   **9**

Latches, Flip flops – SR, JK, T, D, Master/Slave FF, Triggering of FF, Analysis and design of clocked sequential circuits – Design - Moore/Mealy models, state minimization, state assignment, lock - out condition circuit implementation - Counters, Ripple Counters, Ring Counters, Shift registers, Universal Shift Register. Model Development: Designing of rolling display/real time clock

UNIT IV ASYNCHRONOUS SEQUENTIAL CIRCUITS   **9**

Stable and Unstable states, output specifications, cycles and races, state reduction, race free assignments, Hazards, Essential Hazards, Fundamental and Pulse mode sequential circuits, Design of Hazard free circuits.

UNIT V LOGIC FAMILIES AND PROGRAMMABLE LOGIC DEVICES   **9**

Logic families- Propagation Delay, Fan - In and Fan - Out - Noise Margin - RTL ,TTL,ECL, CMOS - Comparison of Logic families - Implementation of combinational logic/sequential logic design using standard ICs, PROM, PLA and PAL, basic memory, static ROM,PROM,EPROM,EEPROM EAPROM.

                    TOTAL NUMBER OF PERIODS :   60

PRACTICAL EXERCISES:

1. Design of adders and subtractors & code converters.
2. Design of Multiplexers & Demultiplexers.
3. Design of Encoders and Decoders.
4. Design of Magnitude Comparators
5. Design and implementation of counters using flip-flops
6. Design and implementation of shift registers.

                      Practical: 30 PERIODS

COURSE OUTCOMES:

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

CO1 Use Boolean algebra and simplification procedures relevant to digital logic.
CO2 Design various combinational digital circuits using logic gates.
CO3 Analyze and design synchronous sequential circuits.
CO4 Analyze and design asynchronous sequential circuits
CO5 Build logic gates and use programmable devices

TEXTBOOKS:

1. Johnson, R.A., Miller, I and Freund J., "Miller and Freund's Probability and Statistics for Engineers", Pearson Education, Asia, 8th Edition, 2015.
2. Milton. J. S. and Arnold. J.C., "Introduction to Probability and Statistics", Tata McGraw Hill, 4th Edition, 2007.
3. Walpole. R.E., Myers. R.H., Myers. S.L. and Ye. K., "Probability and Statistics for Engineers and Scientists", Pearson Education, Asia, 8th Edition, 2007.

REFERENCES:

1. Devore. J.L., "Probability and Statistics for Engineering and the Sciences", Cengage Learning, New   Delhi, 8th Edition, 2014.
2. Papoulis, A. and Unnikrishnapillai, S., "Probability, Random Variables and Stochastic Processes",   McGraw Hill Education India, 4th Edition, New Delhi, 2010.
3. Ross, S.M., "Introduction to Probability and Statistics for Engineers and Scientists", 3rd Edition,   Elsevier, 2004.
4. Spiegel. M.R., Schiller. J. and Srinivasan, R.A., "Schaum's Outline of Theory and Problems of   Probability and Statistics", Tata McGraw Hill Edition, 2004.
5. Gupta S.C. and Kapoor V.K., "Fundamentals of Mathematical Statistics", Sultan Chand & Sons,   New Delhi, 12th Edition, 2020.

Course Code	DATA STRUCTURES USIG C++	L	T	P	C
CS3201		3	0	2	4

COURSE OBJECTIVES:
1. To learn concepts of linear data structures like arrays and linked lists.
2. To grasp the concept of stacks and queues as a linear data structure and theoperations uponthem
3. To explore concepts of hierarchical Tree data structures.
4. To explore optimization strategies for indexing structures and graph algorithms.
5. To understand the concept of searching for quick data retrieval, sorting forarranging data,hash functions strategies for optimized data storage.

12

UNIT I LISTS (Blooms Learning Levels: L3 – Apply)
Abstract Data Types (ADTs) – List ADT – Array-based implementation – Linked list implementation – Singly linked lists – Circularly linked lists – Doubly-linked lists – Applications of lists – Polynomial ADT – Radix Sort – Multilists.

Coding Exercises:
1. Write a C++ programs to implement single linked list ADT to perform following operations.
   a. insert an element into a list.
   b. delete an element from list.
   c. search for a key element in list.
   d. count number of nodes in list.
2. Write a C++ programs to implement doubly linked list ADT to perform following operations.
   a. insert an element into a list at the end.
   b. delete middle element from list.
   c. print the element in reverse order.
Write a C++ programs to Perform Polynomial Addition using singly linked list

Problems-solving Assignments:
1. Using classes, design an online address book to keep track of the names, addresses, phone numbers, and dates of birth of family members, close friends, and certain businessassociates. Your program should be able to handle a maximum of 500 entries. Using linked lists, do the program to handle as many entries as required. Add the following operations:
   a. Add or delete a new entry to the address book.
   b. When the program terminates, write the data in the address book to a disk.

2. Assume that a singly linked list is implemented with a header node, but no tail node, andthat it maintains only a pointer to the header node. Write a class that includes methods to
   a. return the size of the linked list b. print the linked list c. test if a value x is contained inthe linked list d. add a value x if it is not already contained in the linked list e. remove a value x
   if it is contained in the linked list

UNIT II   STACKS AND QUEUES   (Blooms Learning Levels: L3 – Apply)

Stack ADT – Operations – Applications – Balancing Symbols – Evaluating arithmeticexpressions- Infix to Postfix conversion – Function Calls – Queue ADT – Operations – Circular Queue – DeQueue – Applications of Queues.

Coding Exercises:

1. Write a C++ programs to implement a stack operations push, pop, top, and isEmpty using a linked list.
2. Write a C++ programs to implement a queue operations enqueue, dequeue,front,isEmpty using a linked list.
3. Write C++ programs to Convert Infix to Postfix Expression using Stack ADT.

Problems-solving Assignments:

1. Convert the following infix expression into postfix form (A + B)* (C + B)* (ElF).
2. Write a C++ program to check if a given expression with parentheses is balanced using a stack.
3. Write C++ programs to implement a Deque(double ended queue) operationspushFront(), pushBack(), popFront(), popBack(), front(), back(), and isEmpty() using a doubly linked list.

UNIT III   TREES (Blooms Learning Levels: L3 – Apply)
Tree ADT – Tree Traversals – Binary Tree ADT – Expression trees – Binary Search TreeADT – AVL Trees – Splay trees- Red Black Tree -Priority Queue (Heaps) – BinaryHeap.

Coding Exercises:

1. Write C++ programs to implement a binary search tree with the following operations:
    i) Insert an element into a binary search tree.
    ii) Delete an element from a binary search tree.
    iii) Search for a key element in a binary search tree.
2. Write C++ programs that use recursive functions to traverse the given binary tree.
    a) Preorder b) inorder c) postorder.
3. Write C++ programs to implement an AVL Tree.

Problems-solving Assignments:

1. Write a C++ program for a Splay Tree for insertion and search operations.
2. Write a C++ program to implementation the insertion operation for a Red-Black Tree.
3. Write a C++ program a Max-Heap data structure operations insert(), extractMax(), and getMax().

UNIT IV   INDEXING AND GRAPHS(Blooms Learning Levels: L3 – Apply

12

Indexing-B-Tree – B+ Tree. Graph Definition – Representation of Graphs – Types of Graphs – Breadth-first traversal – Depth-first traversal – Bi-connectivity – Euler circuits – Topological Sort – Dijkstra's algorithm – Minimum Spanning Tree – Prim's algorithm – Kruskal'salgorithm.

Coding Exercises:
1. Write a C++ program that demonstrates B-Tree operation -insertion, search, anddisplay.
2. Write a C++ program for Dijkstra's single source shortest path algorithm.
3. Write a C++ program for Prim's Minimum Spanning Tree (MST) algorithm.

Problems-solving Assignments:
1. Write a C++ program that demonstrates Breadth-First Traversal (BFS) in a graphusing an adjacency list representation.
2. Write a C++ program that demonstrates topological sorting using Depth-First Search(DFS) on a directed acyclic graph.
3. Write a C++ program that demonstrates Kruskal's algorithm for finding the MinimumSpanning Tree (MST) of a graph.

UNIT V SEARCHING, SORTING AND HASHING TECHNIQUES

12

Searching – Linear Search – Binary Search. Sorting – Bubble sort – Selection sort – Insertion sort – Shell sort –. Merge Sort – Quick Sort,heap sortgen. Hashing – Hash Functions – Separate Chaining – Open Addressing – Rehashing – Extendible Hashing.

Coding Exercises:
1. Write a C++ programs to implement binary search using a recursive function.
2. Write a C++ program to implement the Insertion Sort algorithm.
3. Write a C++ program to implement the separate chaining technique in hashing.

Problems-solving Assignments:
1. Write a C++ program to implement the linear search algorithm using a non-recursive approach.
2. Write a C++ program to implement the Quick Sort algorithm.
3. Write a C++ program to implement a hash table with rehashing.

TOTAL NUMBER OF PERIODS :   60

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

CO1	Implement linear data structures and legal operations permitted on them.
CO2	Implement stack and queue using array and linked list.
CO3	Use tree data structures and operations permitted on them using C program.
CO4	Apply the indexing and graph concepts and applications to solve different problems.
CO5	Apply a suitable algorithm for searching, sorting and hashing.

TEXTBOOKS:

1. Yashavant Kanetkar, "Data Structures Through C++", 3rd Edition, BPB Publications,2019.
2. Mark Allen Weiss, "Data Structures and Algorithm Analysis in C++", 3rd Edition,Addison Wesley Publishing Company, 2006.

REFERENCES:

1. MT Goodrich, "Data Structures and Algorithms in C++", 2nd Edition, John Wiley& SonsInc,2011.
2. Brijendra Joshi, "Data Structures and Algorithms in C++", 1st Edition, McGrawHill Education, 2010.
3. Sahni Horowitz , "Fundamentals of Data Structures in C++", 2nd Edition,Universities Press, 2008.

Course Code	DATABASE MANAGEMENT SYSTEMS	L	T	P	C
CS3202		3	0	2	4

COURSE OBJECTIVES:

1. To learn the fundamentals of data models and to represent a database system using ER diagrams.
2. To study SQL queries and database programming.
3. To learn the techniques of normalization and functional dependencies.
4. To understand the fundamental concepts of transaction processing- concurrency control techniques and recovery procedures.
5. To have an introductory knowledge about the Storage and Query processing Techniques.

UNIT I INTRODUCTION   **12**

Purpose of Database System – Views of data – Data Models – Database Languages –– Database System Architecture – Database users and Administrator – Entity Relationship model (E-R model) – E-R Diagrams – ER Design Issues-Extended E-R features. Datawarehouse modeling.

Coding Exercises:

1. Construct an E-R diagram for a car insurance company whose customers own one or more cars each. Each car has associated with it zero to any number of recorded accidents. Each insurance policy covers one or more cars and has one or more premium payments associated with it. Each payment is for particular period of time and has an associated due date and date when the payment was received.
2. Design and draw ER diagrams that capture the information of the following schema.
   (a) University Database:
        Instructor, Course Offering Attendance, Student details
   (b) Employment Management System:
        Job, department, Salary/bonus Employee Payroll Qualification

Problem-solving Assignments:

A car rental company maintains a database for all vehicles in its current fleet. For all vehicles, it includes vehicle identification number license number, manufacturer, model, date of purchase and color.

Special data are included for certain types of vehicles.

     Trucks: Cargo capacity
     Sports cars: horsepower, renter age requirement Vans:
     number of passengers
     Off-road vehicles: ground clearance, drive train (four-or two-wheel drive)
     Construct an ER model for the car rental company database.

2. A university registrar's office maintains data about the following entities:
   (1) Courses including code, title, credits, syllabus, and prerequisites;
   (2) Course offerings, including course code, year, semester, section number, instructor(s), timings, and classroom;
   (3) Students, including student-id, name, and program; and
   (4) Instructors, including identification number, name, department, and title. Further, the enrollment of students in courses and grades awarded to students in each course they are enrolled for must be appropriately modeled. Construct an E-R diagram for the registrar's office. Document all assumptions that you make about the mapping constraints.

Relational Algebra, Domain Relational Calculus, Tuple Relational Calculus, DDL Commands - Create, Drop, Alter, Truncate, Rename, Keys - primary Key, Foreign Key, DML Commands - Select, Insert, Update, Delete, Any, All, In, Exists, Non Exists, Union, Intersection, DCL Commands - Grant, Revoke, TCL Commands –Commit, Rollback, Save point, Subqueries - nested, correlated, Joins- Inner, Outer, and Equi, Functions - SUM, COUNT, AVG,MIN,MAX, Clauses - Group By, Having By, Embedded SQL, Dynamic SQL- Creation and Dropping of Views, Creation and Execution of Stored Procedures Cursors and Triggers -Opening, Fetching and Closing, Creation, Insertion, Deletion and Updating.

Coding Exercises:

1. Consider the MOVIE DATABASE:

Movies

Title	Director	Myear	Rating
Fargo	Coen	1996	8.2
Raising Arizona	Coen	1987	7.6
Spiderman	Raimi	2002	7.4
Wonder Boys	Hanson	2000	7.6

Directors

Director	Dyear
Coen	1954
Hanson	1945
Raimi	1959

Actors

Actor	Ayear
Cage	1964
Hanks	1956
Maguire	1975
McDormand	1957

Acts

Actor	Title
Cage	Raising Arizona
Maguire	Spiderman
Maguire	Wonder Boys
McDormand	Fargo
McDormand	Raising Arizona
McDormand	Wonder Boys

Write following relational algebra queries for a given set of relations.
1. Find movies made after 1997
2. Find movies made by Hanson after 1997
3. Find all movies and their ratings
4. Find all actors and directors
5. Find Coen's movies with McDormand

2. Consider a Company database with the following tables:
 EMPLOYEE (Ename, EID, LNAME, FNAME, DOJ, Address, Sex, Salary, Dno)
 DEPARTMENT (Dname, Dnumber, Start date)
Perform the following:
1. Create company database.
2. Viewing all databases.
3. Viewing all Tables in a Database.
4. Creating Tables (With and Without Constraints).
5. Inserting/Updating/Deleting Records in a Table.
6. Saving (Commit) and Undoing (rollback).

3. **Create a table:** DEPARTMENT (DEPTNO, DNAME, LOC)

   Perform the following:
   1. Rename the table dept as department.
   2. Add a new column PINCODE with not null constraints to the existing table DEPT.
   3. Rename the column DNAME to DEPT_NAME in dept table.
   4. Automatically drop all the constraints and views that reference the column DEPT_NAME, along with the column.
   5. Change the data type of column LOC as CHAR with size 10.
   6. Delete the department table.

4. **Create a table:** Employee(empno, empname, dept, salary, DOJ, branch)

   Perform the following
   1. Display all the fields of employee table.
   2. Retrieve employee number and their salary.
   3. Retrieve average salary of all employee.
   4. Retrieve number of an employee.
   5. Retrieve distinct number of an employee.
   6. Retrieve total salary of employee group by employee name and count similar names.
   7. Retrieve empname and salary of employee which is greater than >120000.
   8. Display name of employees in descending order.
   9. Display details of employee whose name is AMIT and salary greater than 50000.

5. **For the given schemas:**

EMPLOYEE(Ename,MNIT,LNAME,No,EDate,Address,Sex,Salary,Dno)
DEPARTMENT(Dname,Dnumber,No,Startdate)

   Perform the String Functions, Date functions and Mathematical functions.

   a. Display the resulting salaries if every employee working on the 'Research' Department is given a 10 percent raise.
   b. Find the sum of the salaries of all employees of the 'Accounts' department, as well as the maximum salary, the minimum salary, and the average salary in this department.
   c. Retrieve the names of employees Controlled by department number 5 (use EXISTS operator).
   d. Retrieve the name of each dept and number of employees working in each department which has at least 2 employees.
   e. Retrieve the names of employees who were born in the 1990s. Retrieve the names of employees and their dept name (using JOIN).

6. **For the given table:** EMPLOYEE (Ename, MNIT, LNAME, No, EDate, Address, gender, Salary, Dno)

   Perform the Following
   i. Creating Views (With and Without Check Option),
   ii. Selecting from a View
   iii. Dropping Views

7. **Write a Pl/SQL program to print integers from 1 to 10 by using PL/SQL FOR loop.**

8. **Given the table:** EMPLOYEE (EmpNo, Name, Salary, Designation, DeptID)
   Write a cursor to select the five highest paid employees from the table.

9. **Given an integer i, write a PL/SQL procedure to insert the tuple (i, 'xxx') into a given relation.**
10. **Create a trigger to insert the updated salary details in employee table.**
11. **Write the SQL queries to implement the following concepts:**
    a. Constraints of all types with more schemas.
    b. Create materialized View Procedure with single schema.

Problem-solving Assignments:
1. EMPLOYEE TABLE with CHECK CONSTRAINT for CITY:
    employee (Empid, Empname, St, City, phone, pin)
   WORKTABLE with DEFAULT CONSTRAINT for COMPANYNAME:
    worktable (empid, company name, salary)
   COMPANY TABLE: - company (companyname, city)
   MANAGER TABLE: - manager (empid, manager name)
   USE REFERENCE INTEGRITY for WORKTABLE and EMPLOYEE tables.
   Write the SQL Queries for the following
    a. Find name, city, phone, pin of the resident of all employees who work for TCS
    b. Find all employee names who don't work for TCS
    c. Find all employees who lives in particular city.
    d. Find the employee who got highest salary company wise.
    e. Find the employee id whose salary is greater than 1 lakh

2. Sales Information System
   A database is being constructed for storing sales information system. A product can be described with a unique product number, product name, price, manufacturer name. The product can sale to a particular client and each client have its own unique client number, client name, client addresses, city, pin code, state and total balance to be required to paid. Each client order to buy product from the salesman. In the order, it has unique sales order number, sales order date, client number, salesman number (unique), billed whole payment by the party or not and its delivery date.
   The salesman has the name, addresses, city, pin code, state, salary of the sales man, delivery date, total quantity ordered, product rate.
   Write the SQL queries for the following –
   a. Rename the column product price of product relation to new_product_rate.
   b. Display the order number and date on which the clients placed their order.
   c. Delete all the records having delivery date before 14thOctober 2015.
   d. Find the sum of products based on manufacturer wise.
   e. List of all orders that were canceled.

UNIT III   RELATIONAL DATABASE DESIGN   12

Functional Dependencies – Non-loss Decomposition – Functional Dependencies – First, Second, Third Normal Forms, Dependency Preservation – Boyce/Codd Normal Form- Multi- valued Dependencies and Fourth Normal Form – Join Dependencies and Fifth Normal Form.

Problems-solving Assignments:

1. Design a database schema for a hospital management system. The system stores informationabout patients, doctors, medical records, and appointments.
   Patients (PatientID, FirstName, LastName, Birthdate, Address, Phone)
   Doctors (DoctorID, FirstName, LastName, Specialty)
   MedicalRecords (RecordID, PatientID, DoctorID, Diagnosis, Prescription)
   Appointments (AppointmentID, PatientID, DoctorID, AppointmentDate)

Normalize this schema up to 2nd Normal Form (2NF) and explain the steps taken during the normalization process. Identify functional dependencies and candidate keys for each table.

2. Design a database schema for a university course registration system. The system has information about students, courses, instructors, and registrations.
   Students (StudentID, FirstName, LastName, Birthdate, address)
   Courses (CourseID, CourseName, Credits, InstructorID)
   Instructors (InstructorID, FirstName, LastName, Department)
   Registrations (RegistrationID, StudentID, CourseID, RegistrationDate)
   Normalize this schema up to Boyce-Codd Normal Form (BCNF) and discuss the advantages of reaching higher normal forms. Identify functional dependencies, candidate keys, and foreign keys as part of the normalization process.

UNIT IV TRANSACTIONS

12

Transaction Concepts - Transaction Recovery – ACID Properties – System Recovery – Media Recovery – Two Phase Commit - Save Points – SQL Facilities for recovery – Concurrency – Need for Concurrency – Locking Protocols – Two Phase Locking – Intent Locking – Deadlock- Serializability – Recovery Isolation Levels – SQL Facilities for Concurrency.

Problems-solving Assignments:

1. Consider the following schedules. The actions are listed in the order they are scheduled, and prefixed with the transaction name.
   S1: T1: R(X), T2: R(X), T1: W(Y), T2: W(Y) T1: R(Y), T2: R(Y)
   S2: T3: W(X), T1: R(X), T1: W(Y), T2: R(Z), T2: W(Z) T3: R(Z)
   For each of the schedules, answer the following questions:
   i) What is the precedence graph for the schedule?
   ii) Is the schedule conflict-serializable? If so, what are all the conflict equivalent serial schedules?
   iii) Is the schedule view-serializable? If so, what are all the view equivalent serial schedules?

2. Consider the following two transactions:

   T1: read(A); read(B)
   If A=0 then B=B+1;
   Write(B)

   T2: read(B); read(A)
   If B=0 then A=A+1
   Write(A)

   Add lock and unlock instructions to transactions T1 and T2, so that they observe two phase locking protocol. Can the execution of these transactions result in deadlock?

3. Give an example of a scenario where two-phase locking leads to deadlock. Prove that Two phase locking guarantees serializability.

UNIT V IMPLEMENTATION TECHNIQUES

Overview of Physical Storage Media – Magnetic Disks – RAID – Tertiary storage – File Organization – Organization of Records in Files - B+ tree Index Files – Static Hashing – Dynamic Hashing –Query Processing Overview – Catalog Information for Cost Estimation – Selection Operation – Sorting – Join Operation – Database Tuning - DataBase Security.

Problems-solving Assignments:
1. "RAID mechanism improves reliability through redundancy". Justify this statement. Also discuss the best situations to be adapted for the different levels of RAID?
2. Construct B+ tree for following data. 30, 31, 23, 32, 22, 28, 24, 29 where number of pointers that fit in one node are 5.

Final Project (sample):
1. Database design and implementation of Pay roll Processing.
2. Database design and implementation of Banking System.
3. Database design and implementation of Movie Ticket Booking System.
4. Database design and implementation of Hospital Management System.
5. Database design and implementation of Job Recruitment Portal.
6. Database design and implementation of Travel Agency System.

TOTAL NUMBER OF PERIODS: 60

COURSE OUTCOMES:

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

CO1	Classify the database applications based on size and complexity.
CO2	Implement SQL queries and database programming.
CO3	Normalize the database and identify the functional dependencies.
CO4	Implement the concept of transaction processing, concurrency control and recovery management.
CO5	Process queries to extract data from a database.

TEXTBOOKS:

1. Abraham Silberschatz, Henry F. Korth, S. Sudharshan,   "Database   System Concepts", Fifth Edition, Tata McGraw Hill, 2006.
2. C.J.Date, A.Kannan, S.Swamynathan, "An Introduction   to Database Systems", Eighth Edition, Pearson Education, 2006.

REFERENCES:

1. Ramez Elmasri, Shamkant B. Navathe, "Fundamentals of Database Systems", Fourth Edition, Pearson Addision wesley, 2007.
2. Raghu Ramakrishnan, "Database Management Systems", Third Edition, McGraw Hill, 2003.
3. S.K.Singh, "Database Systems Concepts, Design and Applications", First Edition, Pearson Education, 2006

Course Code	INTRODUCTION TO JAVA PROGRAMMING	L	T	P	C
CS3203		**3**	**0**	**2**	**4**

COURSE OBJECTIVES:

1. To learn Object Oriented Programming concepts and basic characteristics of Java
2. To know the principles of packages, inheritance and interfaces
3. To define exceptions and use I/O streams
4. To develop a java application with threads and generics classes
5. To design and build simple Graphical User Interface

UNIT I   INTRODUCTION TO OOP AND JAVA FUNDAMENTALS   **12**

Object Oriented Programming - Abstraction – objects and classes - Encapsulation-Inheritance - Polymorphism- OOP in Java – Characteristics of Java – The Java Environment - Java Source File Structure – Compilation Fundamental Programming Structures in Java – Defining classes in Java – constructors, methods-access specifiers - static members - Comments, Data Types, Variables, Operators, Control Flow, Arrays , Packages - JavaDoc comments.

Coding Exercises:
1. Solve problems by using sequential search, binary search, and quadratic sorting algorithms(selection, insertion)
2. Develop stack and queue data structures using classes and objects.

Problems-solving Assignments:
1. Write java program that inputs 5 numbers, each between 10 and 100 inclusive. As each number isread display it only if it's not a duplicate of any number already read display the complete set of unique values input after the user enters each new value.

Coding Exercises:
1. Write a java program to create a super class called Figure that receives the dimensions of two dimensional objects. It also defines a method called area that computes the area of an object. Theprogram derives two subclasses from Figure. The first is Rectangle and second is Triangle. Each of the sub class overridden area() so that it returns the area of a rectangle and a triangle respectively Implement a program that finds the largest element in an array of integers.

Problems-solving Assignments:
1.   Develop a java application with an Employee class with Emp_name, Emp_id, Address, Mail_id, Mobile_no as members. Inherit the classes, Programmer, Assistant Professor, Associate Professor and Professor from employee class. Add Basic Pay (BP) as the member of all the inheritedclasses with 97% of BP as DA, 10 % of BP as HRA, 12% of BP as PF, 0.1% of BP for staff club funds. Generate pay slips for the employees with their gross and net salary.
2. Write a Java Program to create an abstract class named Shape that contains two integers and an empty method named printArea(). Provide three classes named Rectangle, Triangleand Circle such that each one of the classes extends the class Shape. Each one of the classes contains only the methodprintArea( ) that prints the area of the given shape.

UNIT II   INHERITANCE AND INTERFACES   12

Inheritance – Super classes- sub classes –Protected members – constructors in sub classes- the Object class – abstract classes and methods- final methods and classes – Interfaces – defining an interface, implementing interface, differences between classes and interfaces and extending interfaces - Object cloning -inner classes, Array Lists – Strings

Coding Exercises:

1. Write a java program to create a super class called Figure that receives the dimensions of two dimensional objects. It also defines a method called area that computes the area of an object. Theprogram derives two subclasses from Figure. The first is Rectangle and second is Triangle. Each of the sub class overridden area() so that it returns the area of a rectangle and a triangle respectively Implement a program that finds the largest element in an arrayof integers

Problems-solving Assignments:

3.   Develop a java application with an Employee class with Emp_name, Emp_id, Address, Mail_id, Mobile_no as members. Inherit the classes, Programmer, Assistant Professor, Associate Professor and Professor from employee class. Add Basic Pay (BP) as the member of all the inheritedclasses with 97% of BP as DA, 10 % of BP as HRA, 12% of BP as PF, 0.1% of BP for staff club funds. Generate pay slips for the employees with their gross and net salary.

Write a Java Program to create an abstract class named Shape that contains two integers and an empty method named printArea(). Provide three classes named Rectangle, Triangleand Circle such that each one of the classes extends the class Shape. Each one of the classes contains only the methodprintArea( ) that prints the area of the given shape

UNIT III   EXCEPTION HANDLING AND I/O   12

Exceptions - exception hierarchy - throwing and catching exceptions - built in exceptions, creating own exception-User defined Exception. Input / Output Basics – Streams – Byte streams and Characterstreams – Reading and Writing Console – Reading and Writing Files

Coding Exercises:

1. Write a JAVA program divide by zero to cause an ArithmeticException. Use try, catch, and finally blocks to throw an exception, catches the exception and provide a way to handle it. Use finally block regardless of whether an exception occurred or not.

2. Create a user defined a custom exception class named CustomException that inherits from the standard Exception class. This custom exception class has a constructor that takes a message as anargument and passes it to the superclass constructor using the super() call. Here age is the input. The custom exception is intentionally set in a way that the age has a negative value to trigger the custom exception. An appropriate error message is thrown when age is negative. The catch block catches this custom exception and prints the error message.

Problems-solving Assignments:

1. Create a File object representing the file has to be read. Create an input stream (e.g., FileInputStream or BufferedReader) to read data from the file. Use the methods provided by thestream to read data from the file like String line.

UNIT IV   MULTITHREADING AND GENERIC PROGRAMMING   12

Differences between multi-threading and multitasking, thread life cycle, creating threads, synchronizing threads, Inter thread communication, daemon threads, thread groups. Generic Programming – Generic classes – generic methods – Bounded Types – Restrictions and Limitations. Strings: Basic String class, methods ,String Buffer Class & StringBuilder class.

Coding Exercises:
1. Write a Java program that creates three threads. First thread displays "Good Morning" every onesecond, the second thread displays "Hello" every two seconds and the third thread displays "Welcome"every three seconds

Problems-solving Assignments:
1. Write a java program that implements a multi-threaded application that has three threads. First thread generates a random integer every 1 second and if the value is even, the second thread computesthe square of the number and prints. If the value is odd, the third thread will print the value of the cubeof the number.

UNIT V COLLECTIONS FRAMEWORK & DATABASE CONNECTIVITY 12

Collections Framework-Autoboxing -For-Each Style for Loop-Collection Interfaces-Collection Interface-List Interface-Set Interface -Sorted Set Interface-Collection Classes-Array List ClassLinkedList Class-HashSet Class-Linked Hash Set Class-Tree Set Class-Enum Set Class- Accessing a Collection via an Iterator-Using an Iterator-The For-Each Alternative to Iterators-
Storing User-Defined Classes in Collections-Working with Maps-The Map Interfaces-The Map Classes-Arrays- Accessing databases using JDBC connectivity – DAO

Coding Exercises:
1. Create a java program for getting a list of fruits. Print the list using for loop. Add one fruit tothe list. Print the new list.
2. Create a java program for getting a list of numbers. Print the list using for loop. Using map –map the words of the numbers with number like "one" to 1. Print the new list.

Problems-solving Assignments:
1. Develop a program to read employee details from a file, sort them based on salary, and writethe sorted data back to the file
2. Create a program to use JDBC to connect to a database, execute queries, and retrieve data. Create a class JDBCDemo. Use the Database connection details like url, username and password to connect to data base. Load and register the JDBC driver. Establish the connection.Create a statement. Execute a query. Process the result set. Retrieve other columns. Close resources

Final Project (Sample) :
1. Create a project will include creating tasks, adding them to the list, marking them as complete,and displaying the list.

2. Library Management System:
   Build a system to manage books, patrons, and transactions in a library. Users can borrow, return,and search for books, while administrators can manage the catalog and user accounts.

3. Online Shopping System:
   Develop an e-commerce platform where users can browse products, add them to the cart, proceedto checkout, and make payments. You can also incorporate user authentication and order history.

4. Chat Application:
   Create a real-time chat application that allows users to send messages to each other. You canimplement both one-on-one and group chat functionalities using sockets or a client-server architecture.

COURSE OUTCOMES:

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

CO1   Develop Java programs using OOP principles
CO2   Develop Java programs with the concepts inheritance and interfaces
CO3   Build Java applications using exceptions and I/O
CO4   Develop Java applications with threads and generics classes
CO5   Develop interactive Java programs using swings

TEXTBOOKS:

1. Herbert Schildt, "Java: The Complete Reference", 11 th Edition, McGraw Hill Education,New Delhi, 2019
2. Cay S. Horstmann, "Core Java Fundamentals", Volume 1, 11 th Edition, Prentice Hall,2018.

REFERENCES:

1. Paul Deitel Harvey Deitel, Java, How to Program, Prentice Hall; 9th edition, 2018

Course Code	Employability Enhancement Skills – II	L	T	P	C
ES3201	Exploring Mathematical Concepts and Reasoning	0	0	2	1

COURSE OBJECTIVES:

The main objectives of this course are to:

- To categorize, apply and use thought process to understand the concepts of Quantitative methods to enhance problem solving skills.
- To prepare and explain the fundamentals related to various possibilities with numeric ability and probabilities related to quantitative aptitude.
- To critically evaluate numerous possibilities related to puzzles.

UNIT I Time and Distance   9

Introduction – Speed: Late / Early / Usual Time – Average Speed – Chasing – Problems on Train: Crossing Pole, Crossing Platform, Train moving in same and different direction – Boats and Streams: Upstream, Downstream – Clock – Calendar.

UNIT II Probability and Statistics   9

Introduction – Algebra of Events – Addition theorem of Probability – Permutation and Combinations – Problems based on choosing the objects – Statistics: Range – frequency, Arithmetic Mean – Median – Mode – Variance – Standard Deviation – Measures of Dispersion – Coefficient of Variation.

UNIT III Arithmetic and Logical Reasoning   9

Introduction – Mathematical Operations – Blood Relations: Direct, Indirect, coded – Problems on Cubes and Dices: Face identification – Folding and cutting Images – Counting technique of figures – Distance & Direction.

UNIT IV Applied Mathematics   9

Mensuration (2D&3D): Square, Rectangle, Triangle, Circle, Parallelogram, Rhombus, Trapezoid, Quadrilateral, Cube, Cuboid, Cylinder, Cone, Sphere, Miscellaneous – Trigonometry: Ratio, Identities, Heights and Distances – Algebra – Logarithm – Geometry.

UNIT V Verbal and Logical Reasoning   9

Introduction – Venn diagram – Syllogism – Data Sufficiency – Decision Making – Puzzle: Number Puzzle, Letter Puzzle – Ranking Test – Data Arrangement: Linear, Circular, Miscellaneous – Critical Reasoning.

COURSE OUTCOMES:

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

CO1 Use their logical thinking and analytical abilities to solve Quantitative aptitude questions from company specific and other competitive tests.
CO2 Solve questions related to Time etc. from company specific and other competitive tests.
CO3 Illustrate and solve puzzle related questions from specific and other competitive tests.

TEXT BOOKS:

1. Aggarwal, R.S, "Quantitative Aptitude for Competitive Examinations (Revised edition)", S Chand Publishing, 2017.
2. Arun Sharma, "Teach Yourself Quantitative Aptitude", McGraw Hill Education, 2017
3. Aggarwal, R.S, "A Modern Approach to Verbal & Non-Verbal Reasoning" 2nd edition, S Chand Publishing, 2018.

REFERENCES:

1. Akhilesh Khare, "Shortcuts in Mathematics", Createspace Independent Pub, 2015.
2. Ravi Shankar, "Vedic Maths for Competitive Exams", Pustak Mahal, 2016.
3. Abhijit Guha, "Quantitative Aptitude for Competitive Examinations", 7th edition, McGraw Hill Education, 2020.

TOTAL: 30 PERIODS

Course Code	LINEAR ALGEBRA	L	T	P	C
MA3301		**3**	**1**	**0**	**4**

COURSE OBJECTIVES:
1. Tofind the basis and dimension of vector space.
2. To obtain the matrix of linear transformation and its Eigenvalues and eigenvectors.
3. Tofind the orthonormal basis of inner product space and find the least square approximation.
4. Tofind Eigenvalues of a matrix using numerical techniques and perform matrix decomposition.
5. To solve QR and LU decomposition and to learn the applications of linear algebra in computer Science.

UNIT I VECTOR SPACES   **12 (9+3)**

Real and Complex fields - Vector spaces over Real and Complex fields - Subspace - Linear space - Linear independence and dependence - Basis and dimension.

UNIT - II LINEAR TRANSFORMATION   **12 (9+3)**

Linear transformation - Rank space and null space - Rank and nullity - Dimension theorem - Matrix representation of linear transformation – Eigen values and eigenvectors of linear transformation.

UNIT - III INNER PRODUCT SPACES   **12 (9+3)**

Inner product and norms - Properties - Orthogonal, Ortho normal vectors - Gram Schmidt ortho normalization process - Least square approximation.

UNIT - IV EIGEN VALUE PROBLEMS AND MATRIX DECOMPOSITION   **12 (9+3)**

Eigen value Problems: Power method, Jacobi rotation method - Singular valuedecomposition - QR decomposition.

UNIT - V APPLICATIONS OF LINEAR ALGEBRA   **12 (9+3)**

Singular value decomposition and principal component analysis - Introduction to their applications in image processing and machine learning - Coding and Decoding - Least Square solutions.

TOTAL NUMBER OF PERIODS :   60

COURSE OUTCOMES:
After the completion of the course the student will be able to
CO1   Find the basis and dimension of vector space
CO2   Obtain the matrix of linear transformation and its Eigen values and eigenvectors
CO3   Find orthonormal basis of inner product space and find least square approximation.
CO4   Find Eigen values of a matrix using numerical techniques and perform matrix decomposition.
CO5   Learning the applications in Image processing, Machine learning and Cryptography.

TEXTBOOKS:
1. Friedberg A.H, Insel A.J. and Spence L, Linear Algebra, Prentice Hall of India, NewDelhi, 2004.
2. Faires J.D. and Burden R., Numerical Methods, Brooks/Cole (Thomson Publications),New Delhi, 2002.

REFERENCES:

1. Kumaresan S, Linear Algebra - A geometric approach, Prentice Hall of India, New Delhi,Reprint, 2010.
2. Strang G, Linear Algebra and its applications, Thomson (Brooks / Cole) New Delhi, 2005.
3. Gerald C.F. and Wheatley P.O, Applied Numerical Analysis, Pearson Educations, New Delhi,2002.
4. Sundarapandian V, Numerical Linear Algebra, Prentice Hall of India, New Delhi, 2008.
5. Richard Branson, Matrix Operations, Schaum's outline series, 1989
6. Bernard Kolman, David R. Hill, Introductory Linear Algebra, Pearson Educations, New Delhi,First Reprint, 2009.

Course Code	COMPUTER ARCHITECTURE AND ORGANIZATION	L	T	P	C
IT3301		3	0	0	3

COURSE OBJECTIVES:

1. To learn the basic structure and operations of a computer.
2. To learn the arithmetic and logic unit and implementation of fixed-point andfloating point arithmetic unit.
3. To learn the basics of pipelined execution.
4. To understand parallelism and multi-core processors.
5. To understand the memory hierarchies, cache memories and virtual memories.
6. To learn the different ways of communication with I/O devices

UNIT I COMPUTER FUNDAMENTALS 9

Functional Units of a Digital Computer: Von Neumann Architecture – Operation and Operands of Computer Hardware Instruction – Instruction Set Architecture (ISA): Memory Location, Address and Operation – Instruction and Instruction Sequencing – Addressing Modes, Encoding of Machine Instruction – Interaction between Assembly and High-Level Language- MIPS Addressing.

UNIT II COMPUTERS ARITHMETIC AND DATA REPRESENTATION 9

Data Representation, Hardware and software implementation of arithmetic unit for common arithmetic operations: addition, subtraction, multiplication, division(Fixed point and floating point)-floating point IEEE standards and Conversion between integer and real numbers- rounding and truncation;

UNIT III PROCESSOR , CONTROL UNIT & DEVICE SUBSYSTEMS 9

A Basic MIPS implementation — Building a Datapath — Control Implementation Scheme — Pipelining — Pipelined Datapath and control — Handling Data Hazards & Control Hazards — Exceptions.
External storage systems; organization and structure of disk drives Basic I/O controllers such as a keyboard and a mouse;RAID architectures.

UNIT IV PARALLELISIM 9

Parallel processing challenges — Flynn's classification — SISD, MIMD, SIMD, SPMD,and Vector Architectures — Hardware multithreading — multi-core processors and other Shared Memory Multiprocessors — Introduction to Graphics Processing Units, Clusters, Warehouse Scale Computers and other Message-Passing Multiprocessors.

UNIT V Memory System Organization & I/O SYSTEMS 9

Main memory organization, Types of Main memories, and its characteristics and performance; Memory Hierarchy — memory technologies — cache memory — measuring and improving cache performance — virtual memory, TLB's — Accessing I/O Devices — Interrupts — Direct Memory Access — Bus structure — Bus operation — Arbitration — Interface circuits — USB.I/Ofundamentals: handshaking, buffering; I/O techniques: programmed I/O, interrupt-driven I/O, DMA .

TOTAL NUMBER OF PERIODS : 45

COURSE OUTCOMES:

After the completion of the course the student will be able to

CO1 Understand the basics structure of computers, operations and instructions.
CO2 Design arithmetic and logic unit.
CO3 Understand pipelined execution and design control unit
CO4 Understand parallel processing architectures.

CO5   Understand the various memory systems and I/O communication.

TEXTBOOKS:

1. John L. Hennessey & D.A. Patterson, Computer architecture: A quantitative approach, ,
   Fifth
   Edition, Morgan Kaufman, // Elsevier Publishers, Fifth Edition, 2012.
2. David A. Patterson and John L. Hennessy, Computer Organization and Design: The
   Hardware/Software Interface, Fifth Edition, Morgan Kaufmann / Elsevier, 2014.
3. Carl Hamacher, Zvonko Vranesic, Safwat Zaky and Naraig Manjikian, Computer
   Organization and Embedded Systems, Sixth Edition, Tata McGraw Hill, 2012.

REFERENCES:

1. William Stallings, Computer Organization and Architecture – Designing for Performance,
   EighthEdition, Pearson Education, 2010.
2. John P. Hayes, Computer Architecture and Organization, Third Edition, Tata McGraw Hill,
   2012.
3. M. M. Mano, Computer System Architecture, Third Edition, Prentice-Hall 2008.

Course Code	DESIGN AND ANALYSIS OF ALGORITHMS	L	T	P	C
IT3302		3	0	2	4

COURSE OBJECTIVES:
1. To understand and apply the algorithm analysis techniques on searching and sorting algorithms
2. To critically analyze the efficiency of graph algorithm.
3. To understand different algorithm design techniques
4. To solve programming problems using state space tree
5. To understand the concepts behind NP Completeness, Approximation algorithms andrandomize algorithms.

UNIT I INTRODUCTION   9

Problem Solving : Programs and Algorithms-Problem Solving Aspects-Problem Solving Techniques-   Algorithm analysis: Time and space complexity - Asymptotic Notations and its properties Best case, Worst case and average case analysis– Recurrence relation: substitution method – **String Matching and** searching: Interpolation Search, Pattern search: The naïve string- matching algorithm - Rabin-Karp algorithm - Knuth-Morris-Pratt algorithm.

.

UNIT - II GREEDY TECHNIQUE   9

Minimum spanning tree, Kruskal's and Prim's algorithm- Shortest path: Bellman-Ford algorithm - Dijkstra's algorithm - Floyd-Marshall algorithm - Network flow : Flow networks - Ford-Fulkerson method – Maximum bipartite matching.

UNIT - III DIVIDE-AND-CONQUER AND DYNAMIC PROGRAMMING   9

Divide and Conquer methodology: Finding maximum and minimum - Merge sort - Quick sort Dynamic programming: Elements of dynamic programming — Matrix-chain multiplication - Multi stage graph — Optimal Binary Search Trees - a c t ivity-selection problem –- Optimal Merge pattern-0/1 Knap sack problem.

UNIT - IV STATE SPACE SEARCH AND BACKTRACKING   9

Backtracking: n-Queens problem - Hamiltonian Circuit Problem - Subset Sum Problem – Graph colouring problem Branch and Bound: Solving 15-Puzzle problem - Assignment problem - Knapsack Problem - Travelling Salesman Problem.

UNIT - V NP-COMPLETE AND APPROXIMATION ALGORITHM   9

Tractable and intractable problems: Polynomial time algorithms – Venn diagram representation - NP- algorithms - NP-hardness and NP-completeness – Bin Packing problem - Problem reduction: TSP – 3- CNF problem. Approximation Algorithms: TSP - Randomized Algorithms: concept and application - primality testing - randomized quick sort - Finding kth smallest number.

TOTAL PERIODS: 45

Assignment

Assignment 1: Algorithm Analysis and Time Complexity

Title: Asymptotic Analysis of Sorting Algorithms

Objective: Compare the time complexity of two sorting algorithms - Insertion Sort and HeapSort.

Steps:
1. Provide a brief overview of Insertion Sort and Heap Sort algorithms.
2. Analyze the time complexity of each algorithm in terms of Big O notation.
3. Discuss the best-case, worst-case, and average-case scenarios for both algorithms.
4. Compare the efficiency of Insertion Sort and Heap Sort based on their time complexities.
5. Discuss any trade-offs or advantages one algorithm may have over the other.

Assignment 2: String Matching Algorithms

Title: Comparative Analysis of String-Matching Algorithms

Objective: Analyze and compare the string-matching algorithms - Naïve String Matching,Rabin-Karp, and Knuth-Morris-Pratt.

Steps:
1. Provide a brief overview of each string-matching algorithm.
2. Analyze the time complexity of Naïve String Matching, Rabin-Karp, and Knuth-Morris-Pratt.
3. Discuss the best-case, worst-case, and average-case scenarios for each algorithm.
4. Compare the strengths and weaknesses of the algorithms, considering factors likepattern length and text size.

Provide examples or scenarios where each algorithm may be most suitable.

Assignment 3: Divide and Conquer and Dynamic Programming

Title: Comparative Study of Divide and Conquer vs. Dynamic Programming

Objective: Analyze and compare the efficiency of Divide and Conquer and Dynamic Programming approaches in solving two different problems - Finding Maximum and Minimum, and Matrix-Chain Multiplication.

Steps:
1. Finding Maximum and Minimum:
   - Explain the Divide and Conquer approach for finding the maximum andminimum in an array.
   - Analyze the time complexity of the Divide and Conquer solution.
   - Discuss any limitations or scenarios where this approach might be preferable.
2. Matrix-Chain Multiplication:
   - Describe the Dynamic Programming approach for solving the Matrix-ChainMultiplication problem.
   - Analyze the time and space complexity of the Dynamic Programming solution.
   - Discuss the advantages of using Dynamic Programming in this context.
3. Comparison:
   - Compare the Divide and Conquer and Dynamic Programming approachesin terms of time and space complexity.

Discuss scenarios where one approach might be more suitable than the other.

TOTAL: 75 PERIODS

COURSE OUTCOMES:

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

CO1: Understand the fundamental principles of problem solving and analyse the efficiency of algorithms using various frameworks

CO2: Apply Greedy Technique to Solve real time problems
CO3: Implement and evaluate various Divide and Conquer and Dynamic Programming Techniques
CO4: Apply Backtracking techniques to solve real-world problems.
CO5: Evaluate and Solve problems using approximation algorithms and randomized algorithms

TEXTBOOKS:

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, "Introduction to Algorithms", 3rd Edition, Prentice Hall of India, 2009.
2. Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran "Computer Algorithms/C++" Orient Blackswan, 2nd Edition, 2019.

REFERENCES:

1. Anany Levitin, "Introduction to the Design and Analysis of Algorithms", 3rd Edition, PearsonEducation, 2012.
2. Alfred V. Aho, John E. Hopcroft and Jeffrey D. Ullman, "Data Structures and Algorithms", ReprintEdition, Pearson Education, 2006.
3. S. Sridhar, "Design and Analysis of Algorithms", Oxford university press, 2014.

PRACTICAL EXERCISES:

A. Searching and Sorting Algorithms

1. Implement Linear Search. Determine the time required to search for an element. Repeat the experiment for different values of n, the number of elements in the list to be searched and plot a graph of the time taken versus n.

2. Implement recursive Binary Search. Determine the time required to search an element. Repeat the experiment for different values of n, the number of elements in the list to besearched and plot a graph of the time taken versus n.

3. Given a text txt [0...n-1] and a pattern pat [0...m-1], write a function search (char pat [ ], char txt [ ]) that prints all occurrences of pat [ ] in txt[ ]. You may assume that n>m.

4. Sort a given set of elements using the Insertion Sort and Heap sort methods and determine the time required to sort the elements. Repeat the experiment for different values of n, the number of elements in the list to be sorted and plot a graph of the time taken versus n.

B. Graph Algorithms

1.Develop a program to implement graph traversal using Breadth First Search.

2.Develop a program to implement graph traversal using Depth First Search.

3. From a given vertex in a weighted connected graph, develop a program to find the shortest paths to other vertices using Dijkstra's algorithm.

4.Find the minimum cost spanning tree of a given undirected graph using Prim's algorithm.

5. Implement Floyd's algorithm for the ALL-Pair Shortest path problem

6.. Compute the transitive closure of a given directed graph using Marshall's algorithm.

C. Algorithm Design Techniques

1. Develop a program to find out the maximum and minimum numbers in a given list of n numbers using   the divide and conquer technique.

2. Implement Merge sort and Quick sort methods to sort an array of elements and determine the time required to sort. Repeat the experiment for different values of n, the number of elements in the list to be sorted and plot a graph of the time taken versus n.

D. State Space Search Algorithms

1. Implement N Queens problem using Backtracking.

E. Approximation Algorithms Randomized Algorithms

1. Implement any scheme to find the optimal solution for the Traveling Sales person problem and then solve the same problem instance using any approximation algorithm and determine the error in the approximation.

2. Implement randomized algorithms for finding the kth smallest number. The programs can be implemented in C/C++/JAVA/Python.

Practicals:30 Periods

Course Code	WEB DEVELOPMENT PRACTICES	L	T	P	C
IT3303		3	0	2	4

COURSE OBJECTIVES:

1. To understand the difference between web based and server-side programming.
2. To create awareness on web technology.
3. To create full stack application using java script-based frameworks.
4. To develop Python based innovative full stack applications.
5. To develop java applications using Spring boot and Hibernate.

UNIT I – NODE.JS 9

Introduction to Server-side programming – multi-tier architecture - Node.js architecture – npm – Development environment – API.

UNIT II – EXPRESS.JS 9

Express JS and features – Routing – HTTP request and response – Middleware – Error Handling.

UNIT III – PYTHON FRAMEWORKS 9

Introduction to frameworks on Python – Flask and Django. Django: Creating web application – handle request and response – views and templates – Forms and generic views – SQLAlchemy.

UNIT IV – MONGODB 9

SQL and NoSQL concepts – Create and manage MongoDB – Migration of Data into MongoDB – MongoDB with Node.js – Services offered by MongoDB.

UNIT V – JAVA DEVELOPMENT ENVIRONMENT 9

Introduction to Spring Boot – Creating Project – Spring Initializer – Download and Install STS IDE – Spring Boot Example – Spring Boot CLI - Spring Boot Annotations - Spring Boot Application Properties - Spring Boot Starter Web - Spring Boot AOP Hibernate – Hibernate Architecture - Hibernate with eclipse – Hibernate web application example – Hibernate Log4j – Hibernate Inheritance Mapping – Hibernate and Spring Integration.

TOTAL PERIODS :   45

PRACTICAL EXERCISES: **30 PERIODS**
1. Project – Simple weather application using MongoDB
2. Project – URL shortener Application using SQL.
3. Project - Flight Ticket Booking

Create a web application for flight ticket booking. Use any tech stack for the backend and db.Aconsole-based application would work. Submissions with a very basic UI is mandatory.

Type of Users
a.   User
b.   Admin

User Use Cases
● Login
● Sign up
● Searching for flights based on date and time
● Booking tickets on a flight based on availability (assuming the defaultseat count is 60)
● My Booking -> to list out all the bookings made by that user
● Logout

Admin Use Cases
● Login (Seperate login for Admin)
● Add Flights
● Remove flights
● View all the booking based on flight number and time

COURSE OUTCOMES:
After the completion of the course the student will be able to
CO1   Find the basis and dimension of vector space
CO2   Obtain the matrix of linear transformation and its Eigen values and eigenvectors
CO3   Find orthonormal basis of inner product space and find least square approximation.
CO4   Find Eigen values of a matrix using numerical techniques and perform matrix decomposition.
CO5   Learning the applications in Image processing, Machine learning and Cryptography.

                                                          TOTAL: 75 PERIODS

TEXTBOOKS:
1. Jonathan Wexler, "Get Programming with Node.js", Manning Publications, 2019.
2. Beginning Node.js, Express & MongoDB Development, Greg Lim, 1st Edition, 2019.

ONLINE RESOURCES:
1. Django documentation - https://docs.djangoproject.com/en/4.2/
2. MongoDB documentation - https://www.mongodb.com/docs/
3. Spring boot documentation - https://spring.io/guides/gs/spring-boot/
4. Hibernate documentation - https://hibernate.org/

Course Code	COMPUTER NETWORKS	L	T	P	C
IT3304		3	0	2	4

COURSE OBJECTIVES:

The main objectives of this course are to:

- Understand Fundamental Networking Concepts
- Explore Advanced Data Link and Network Layer Technologies.
- Enhance Knowledge of Transport and Application Layer Protocols.
- Develop Proficiency in Network Security and Emerging Technologies.
- Hands-On Experience through Simulations and Experiments.

UNIT I - INTRODUCTION TO NETWORKING   **9**

Overview of Computer Networks: Definition and importance of computer networks - Historical development of networking, Layering in Network Architectures: OSI model overview - TCP/IP protocol stack - Functions and responsibilities of each layer, Switching Techniques: Packet switching basics - Circuit-switching and virtual circuit-switching concepts, Data Link Layer: Framing techniques - Error detection methods - Medium Access Control (MAC) protocols - Ethernet bridging

UNIT II - ROUTING AND IP ADDRESSING   **9**

Routing Protocols: Shortest path routing - Flooding algorithm - Distance vector routing - Link state routing, Fragmentation and IP Addressing: Basics of IP addressing - IPv4 addressing - CIDR notation - Fragmentation in network communication, IP Support Protocols: Address Resolution Protocol (ARP) - Dynamic Host Configuration Protocol (DHCP) - Internet Control Message Protocol (ICMP) - Network Address Translation (NAT)

UNIT III - TRANSPORT LAYER   **9**

Flow Control and Congestion Control: Basics of flow control - Congestion control mechanisms, Transmission Control Protocol (TCP): TCP features and functionalities, TCP connection establishment, maintenance, and termination, User Datagram Protocol (UDP): Characteristics and usage scenarios - Comparison with TCP, Sockets: Overview of sockets in network programming

UNIT IV - APPLICATION LAYER PROTOCOLS   **9**

Domain Name System (DNS): DNS fundamentals - DNS resolution process, Simple Mail Transfer Protocol (SMTP): Email communication basics - SMTP operation, Hypertext Transfer Protocol (HTTP): Basics of web communication, HTTP request-response model, File Transfer Protocol (FTP): FTP modes and operation, Security considerations in FTP

UNIT V NETWORK SECURITY AND EMERGING TECHNOLOGIES   **9**

Network Security Technologies: Intrusion detection systems, firewalls, VPNs. Blockchain Technology in Networking: Basics of blockchain, applications in networking. Zero Trust Networking: Concepts and implementation strategies

45 PERIODS

PRACTICAL EXERCISE

1. Learn to use commands like tcp dump, netstat,ifconfig, nslookup and trace route. Capture ping and trace route PDUs using a network protocol analyzer and examine.
2. Write a HTTP web client program to download a web page using TCPsockets.
3. Applications using TCP sockets like: a) Echo client and echo server   b)Chat
4. Simulation of DNS using UDP sockets.
5. Use a tool like Wireshark to capture packets and examine the packets
6. Write a code simulating ARP/RARP protocols.
7. Study of Network simulator (NS) and Simulation of Congestion Control Algorithms using NS.
8. Study of TCP/UDP performance using Simulation tool.
9. Simulation of Distance Vector/Link State Routing algorithm.
10. Simulation of an error correction code(like CRC**)**   **TOTAL75 PERIODS**

Course Format

Lectures and discussions, Hands-on coding exercises and projects, Guest lectures by industry Experts, Group discussions and presentations, Online resources and tutorials

---

Assessments & Grading

Quizzes / Assignments, Project, 3 Internal Assessments, Final Examination

---

COURSE OUTCOMES:
After the completion of the course the student will be able to
CO1   Apply the fundamentals of communication in networking aspects
CO2   Analyze the various protocols in Physical, Data link, Network, Transport and Application layers and their mechanisms.
CO3   Design functional aspects for network applications.
CO4   Develop programs that demonstrate the operations of physical, Data Link, Network, Transport layers**.**
CO5   Develop programs that demonstrate the operations Application layers**.**

TEXTBOOKS:
1. Data Communication and Networking, 4th Edition, Behrouz A. Forouzan, McGrawHill.
2. Data and Computer Communication, 8th Edition, William Stallings, Pearson Prentice Hall India.
3. Computer Networks – A Systems Approach, 4th Edition, Larry L. Peterson and Bruce S. Davie, Elsevier

REFERENCES:
1. Computer Networks, 8th Edition, Andrew S. Tanenbaum, Pearson New International Edition. Course Code: BTCS 504-18 Course Title: Computer Networks 3L:1T:0P 3Credits 42 Hours
2. Internetworking with TCP/IP, Volume 1, 6th Edition Douglas Comer, Prentice Hall of India.
3. TCP/IP Illustrated, Volume 1, W. Richard Stevens, Addison-Wesley, United States of America.

E-Book:

An Introduction to Computer Networks Peter L Dordal First - 2020
http://intronetworks.cs.luc.edu
/current/ComputerNetworks

Course Code	PYTHON PROGRAMMING	L	T	P	C
IT3305		3	0	2	4

COURSE OBJECTIVES:

1. The course demonstrates an in-depth understanding of the Python language with basic syntax and semantics.
2. To learn advanced Python programming features to develop GUI applications and access databases and object-oriented concepts.
3. To introduce the libraries for python programming.
4. To introduce the basics of parallel computing and Memory IO communication in Python.
5. To Learn GPU programming and Parallel AI Computing.

UNIT I BASIC SYNTAX 9

Python Interpreter, Execution model, Interactive prompt and IDLE, Basic syntactical structures, Identifiers and keywords, Basic data types – integers, Booleans, float, complex, Decimal, other standard library types, String type. String formatting, built-in functions, simple programs.

Sequence types – Lists, Tuples, Named tuples, Set types -- Sets, Frozen sets, Mapping types

– Dictionaries, Default dictionaries, iterating collections, Copying collections, Comprehensions, Generators, Control structures – Conditional branching, Looping, Exception handling.

UNIT II PROGRAMMING STRUCTURES 9

Functions – types of functions, scopes, global and nonlocal statements, argument passing techniques, argument and parameter unpacking, Assertions, and Recursion. Standard library modules – os, sys, string, math, calendar, time, date time, HTML, Custom modules and packages.

OOP - Custom classes, Inheritance and Polymorphism, Properties, Custom collection classes, Decorators, Context managers, Meta-classes. File handling -- Reading and writing binary and text data, command-line arguments. Structured text files – CSV, XML, HTML, JSON. Debugging and testing, frameworks. The Django web framework – usage and example.

UNIT III FUNCTIONS, MODULES AND PACKAGES 9

Basics of Numpy arrays –aggregations –computations on arrays –comparisons, masks, boolean logic – fancy indexing – structured arrays – Data manipulation with Pandas – data indexing and selection. Scientific computing and numerical simulations with SciPy and SimPy. Large-scale data analysis and machine learning with Pandas, Scikit-learn, and TensorFlow/PyTorch

UNIT IV PARALLEL COMPUTING BASICS 9

Introduction to Parallel Programming - CPU, cores, threads, and Processes- concurrent and parallel models - GIL, Memory Organization, and Communication - Building Multithreaded Programs - Working with Multiprocessing and mpi4py Library - Asynchronous Programming with AsyncIO. Realizing Parallelism with Distributed Systems.

UNIT V ADVANCED PARALLEL COMPUTING **9**

Maximizing Performance with GPU Programming using CUDA - Embracing the Parallel Computing Revolution Scaling Your Data Science Applications with Dask - Exploring the potential of AI with Parallel Computing Hands-on Applications of Parallel Computing.

TOTAL PERIODS :45

LIST OF LAB EXPERIMENTS **30 PERIODS**

1 Function Types and Scopes Analysis: Analyse the scopes of different types of functions (global, local, nonlocal) using print statements or debugging tools like pdb.

2 Argument Passing Techniques Comparison: Compare different argument passing techniques (e.g., positional, keyword, default, variable-length) in Python functions through experimental evaluation.

3 Parameter Unpacking Demonstration: Create experiments to demonstrate the unpacking of function arguments using *args and **kwargs.

4 Assertion Testing: Explore the usage and effectiveness of assertions in debugging Python code by designing experiments with custom assertions.

5 Recursion Performance Analysis: Measure and compare the performance of recursive functions with iterative solutions for various problem sets.

6 Custom Classes and Inheritance: Experiment with custom class inheritance hierarchies to understand the behavior of inherited methods and attributes.

7 Polymorphism Demonstration: Design experiments to demonstrate polymorphic behavior in Python classes using method overriding and overloading.

8 Properties vs Direct Access: Compare the performance of accessing class attributes directly versus using properties for data encapsulation.

9 Custom Collection Classes Evaluation: Implement and evaluate custom collection classes in terms of performance and usability compared to built-in Python data structures.

10 Decorators Impact on Performance: Investigate the overhead introduced by decorators on function execution time and memory consumption.

11 File Handling Efficiency Study: Compare the efficiency of reading and writing binary and text data using different file handling techniques.

12 Structured Text File Parsing: Experiment with parsing CSV, XML, HTML, and JSON files to understand their structure and performance implications.

13 Debugging and Testing Frameworks Comparison: Compare the effectiveness and ease of use of different debugging and testing frameworks for Python.

14 Django Web Framework Usage: Develop a simple web application using Django and evaluate its performance under various loads and scenarios.

15 Parallel Computing Performance Evaluation: Measure the performance improvement achieved by parallel computing techniques using CPU cores, threads, and multiprocessing compared to sequential execution for specific tasks.

COURSE OUTCOMES:

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

CO1   Mastery of Python's syntax and semantics for robust programming.

CO2   Proficiency in advanced Python for GUIs, databases, and OOP.

CO3   Familiarity with essential Python libraries for diverse applications.

CO4   Understanding of parallel computing basics and Memory IO communication.

CO5   Acquisition of skills in GPU programming and Parallel AI Computing.

TEXTBOOKS:

1. "Think Python: How to Think Like a Computer Scientist", Allen B . Downey, 2nd edition, Shroff/O'Reilly Publishers, 2016.

2. "An Introduction to Python", Guido van Rossum and Fred L. Drake Jr – Network TheoryLtd., 2011.

3. Nelli, F. (2023). Parallel and high performance programming with Python: Unlock parallel and concurrent programming in Python using multithreading, CUDA, Pytorch and DASK. Orange Education Pvt.

REFERENCES:

1. "Introduction to Computation and Programming Using Python", John V Guttag Revisedand Expanded Edition, MIT Press, 2013

2. "Introduction to Programming in Python: An Inter-disciplinary Approach", Robert Sedgewick, Kevin Wayne, Robert Dondero, Pearson India Education Services Pvt. Ltd., 2016.

3. Exploring Python‖, Timothy A. Budd, Mc-Graw Hill Education (India) Private Ltd., 2015.

4. Fundamentals of Python: First Programs‖, Kenneth A. Lambert, CENGAGE Learning,2012.

5. Programming Prowess: Conquering 110 Coding Challenges Illustrated with Python Code. M. Sabrigiriraj, K. Monoharan, ISBN: 978-81-969837-3-4. 2024

Course Code	Employability Enhancement Skills – III	L	T	P	C
ES3201	Professional Communication and Teamwork Skills	0	0	2	1

COURSE OBJECTIVES:

The main objectives of this course are to:

- Tofamiliarize students with various forms of communication.
- To develop effective team communication skills.
- To enhance stakeholder communication skills.
- To cultivate ethical communication practices.
- To explore digital communication tools and trends.

UNIT I 9

Introduction to Communication - Verbal Communication Skills: - Written Communication Skills - Nonverbal Communication - Interpersonal Communication.

UNIT II 9

Characteristics of Effective Teams - Team Building and Group Cohesion - Conflict Resolution - Decision Making in Teams - Cross-Cultural Communication.

UNIT III 9

Stakeholder Communication - Presentation Skills - Effective Meetings - Feedback and Evaluation.

UNIT IV 9

Professional Codes of Conduct - Integrity in Communication - Addressing Ethical Challenges - Analyzing real-world ethical communication dilemmas.

UNIT V 9

Digital Communication Tools - Social Media and Networking - Emerging Trends in Communication.

COURSE OUTCOMES:

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

CO1: Demonstrate proficiency in various forms of communication.
CO2: Exhibit strong team communication skills.
CO3: Display competence in stakeholder communication.
CO4: Apply ethical communication principles.
CO5: Utilize digital communication tools effectively.

TEXT BOOKS:

1. Sharon J. Gerson and Steven M. Gerson. "Technical Communication: Process and Product", Pearson, 2014
2. Karl A. Smith. "Teamwork and Project Management", McGraw-Hill Education, 2013
3. Charles E. Harris Jr., Michael S. Pritchard, and Michael J. Rabins. "Engineering Ethics: Concepts and Cases", Cengage Learning, 2012
4. Christoph Meinel and Harald Sack. "Digital Communication: Communication, Multimedia, Security", Springer, 2014.

REFERENCES:

1. Katherine L. Adams and Gloria J. Galanes. "Communicating in Groups: Applications and Skills", McGraw Hill Education, 2018
2. Lawrence Holpp. "Managing Teams: Strategies for Success", McGraw Hill, 1998.
3. Caroline Whitbeck (ed) "Ethics in Engineering Practice and Research", Cambridge University Press, 2011.

TOTAL: 30 PERIODS

Course Code	DISCRETE MATHEMATICS	L	T	P	C
MA3402		3	1	0	4

COURSE OBJECTIVES:
1. To familiarize the applications of algebraic structures
2. To introduce most of the basic terminologies used in computer science courses and application of ideas to solve practical problems
3. To understand the graph models and basic concepts of graphs
4. To study the characterization and properties of trees and graph connectivity
5. To extend student's logical and mathematical maturity and ability to deal with abstraction

UNIT I Algebraic Structures   **12 (9+3)**

Semigroup – Monoids – Groups – Subgroups – Abelian groups – Lagrange's theorem – Rings (examples only) – Integral domain – Fields – Definition and examples.

UNIT II Combinatorics   **12 (9+3)**

Introduction to Basic Counting Principles – Permutations and combinations (nPr, nCr) – Balls and Pins problems – Pigeonhole Principle – Recurrence relations – Generating Functions - Introduction to Proof Techniques – Mathematical Induction.

UNIT III Sets, Relations, Functions, and Basic Graph Theory   **12 (9+3)**

Sets: Basic definitions and operations – Relations: Properties of relations, Equivalence relations, Partial orders and lattices – Function: Definitions, types of functions, composition, and inverse functions – Graphs & Digraphs; Definition & basic properties, Isomorphism, complement – Connectivity and Reachability: Adjacency matrix – Matching & Colouring: Basics concepts of graph matching, Chromatic number, graph colouring problem – Eulerian & Hamiltonian Paths & Circuits.

UNIT IV Trees and Planar Graphs   **12 (9+3)**

Trees – Planar graphs, Euler's formula, dual of a planer graph – Independence number and clique number, chromatic number, statement of Four-colour theorem.

UNIT V Propositional Logic and Boolean Algebra
12 (9+3)
Propositional calculus – propositions and connectives, syntax – Semantics: truth assignments and truth tables, validity and satisfiability, tautology – Adequate set of connectives – Equivalence and normal forms; Compactness and resolution – Formal reducibility: natural deduction system and axiom system – Soundness and completeness – Introduction of Boolean algebra, truth table, basic logic gate, basic postulates of Boolean algebra, principle of duality, canonical form, Karnaugh map.

                TOTAL NUMBER OF PERIODS :   60

COURSE OUTCOMES:
After the completion of the course, the students will be able to:
CO1   Have an understanding in identifying structures on many levels.
CO2   Understand the concepts of combinatorics.
CO3   Understand the concepts of graph theory and its applications.

CO4   Understand the importance of the natural applications of trees and apply the graph coloring

    concepts in partitioning problems

CO5   Learning logic and Boolean algebra and using these concepts to solve the problems.

TEXTBOOKS:

  1. Rosen. K.H., "Discrete Mathematics and its Applications", 7th Edition, Tata McGraw Hill Pub.Co. Ltd., New Delhi, Special Indian Edition, 2017

2. Tremblay. J.P. and Manohar. R, "Discrete Mathematical Structures with Applications to ComputerScience", Tata McGraw Hill Pub. Co. Ltd, New Delhi, 30th Reprint, 2011

3.

REFERENCES:

1.Grimaldi. R.P. "Discrete and Combinatorial Mathematics: An Applied Introduction", 5th Edition,Pearson Education Asia, Delhi, 2013.

2. Koshy. T. "Discrete Mathematics with Applications", Elsevier Publications, 2006

3. Lipschutz. S. and Mark Lipson., "Discrete Mathematics", Schaum's Outlines, Tata McGraw HillPub. Co. Ltd., New Delhi, 3rd Edition, 2010

4. Gilberft Strang, "Introduction to Linear Algebra".

5. R. A. Brualdi, "Introductory Combinatorics", , North-Holland, New York.

6. N. Deo, "Graph Theory with Applications to Engineering and Computer Science", Prentice Hall,Englewood Cliffs.

Course Code	COMPUTING THEORY AND COMPILER DESIGN	L	T	P	C
IT3401		3	0	0	3

COURSE OBJECTIVES:

- Construct automata for any given pattern and find its equivalent regular expressions
- Understand different phases of the compiler and various parsing techniques
- Learn how to generate machine codes.
- Understand Turing machines and basics of theory of computation
- Understand undecidability and semi-decidability

UNIT – I Formal Language and Regular Expressions   9

Languages, Definition Languages regular expressions, Finite Automata – DFA, NFA. Conversion of regular expression to NFA, NFA to DFA. Applications of Finite Automata to lexical analysis, lex tools – pumping lemma.

UNIT – II Context Free grammars and Parsing   9

Context free grammars, derivation, parse trees, ambiguity LL(K) grammars and LL(1) parsing Bottom up parsing, handle pruning, LR Grammar Parsing, LALR parsing, parsing ambiguous grammars, YACC programming specification – Pushdown Automata – pumping lemma.
Turing Machine - Undecidability

UNIT – III Syntax-Directed Translation and Intermediate-Code Generation   9

Syntax directed translation, S-attributed and L-attributed grammars, Intermediate code Generation – abstract syntax tree, translation of simple statements and control flow statements. Context Sensitive features – Chomsky hierarchy of languages and recognizers. Type checking, type conversions, equivalence of type expressions, overloading of functions and operations.

UNIT – IV Run-Time Environments and Code Generation   9

Storage organization, Storage Allocation Strategies: Static, Stack and Heap allocation – Parameter Passing – Symbol Tables – Dynamic Storage Allocation – Issues in the design of a code generator – Basic Blocks and Flow graphs – Design of a simple Code Generator – Optimal Code Generation for Expressions – Dynamic Programming Code Generation

UNIT – V Code Optimization   9

Principal sources of Optimization – Peep-hole optimization – DAG - Optimization of Basic Blocks – Global Data Flow Analysis – Efficient Data Flow Algorithm – Recent trends in Compiler Design.

TOTAL:45 PERIODS

COURSE OUTCOMES:

On successful completion of this course, the students will be able to

1. Construct automata, regular expression for any pattern.
2. Write Context free grammar for any construct.
3. Build the different Phases of compiler and apply the various optimization techniques.
4. Design Turing machine for a given language
5. Explain decidability, semi-decidability, and undecidability

TEXTBOOKS:

1. John E Hopcroft and Jeffery D Ullman, Introduction to Automata Theory, Languages and Computations, Narosa Publishing House, 2002.

2. Alfred V Aho, Monica S. Lam, Ravi Sethi and Jeffrey D Ullman, Compilers – Principles, Techniques and Tools, 2nd Edition, Pearson Education, 2007.

REFERENCES:

1. Michael Sipser, "Introduction of the Theory of Computation", Second Edition, Thomson Brokecole, 2006.

2. J. Martin, "Introduction to Languages and the Theory of Computation", Third Edition, Tata McGraw Hill, 2003.

3. V. Raghavan, Principles of Compiler Design, Tata McGraw Hill Education Publishers, 2010.

4. Muneeswaran. K, —Compiler Design, Oxford University Press, 2012

5. Steven S. Muchnick, —Advanced Compiler Design and Implementation, Morgan Kaufmann Publishers - Elsevier Science, India, Indian Reprint 2003.

6. Randy Allen, Ken Kennedy, —Optimizing Compilers for Modern Architectures: A Dependence-based Approach, Morgan Kaufmann Publishers, 2002.

Course Code	OPERATING SYSTEMS	L	T	P	C
IT3402		3	0	2	4

COURSE OBJECTIVES:

1. To understand the basics and functions of operating systems
2. To understand processes and threads
3. To analyze scheduling algorithms and process synchronization
4. To understand the concept of deadlocks
5. To analyze various memory management schemes
6. To be familiar with I/O management and file systems
7. To be familiar with the basics of virtual machines and Mobile OS like iOS and Android.

UNIT I INTRODUCTION   9

Computer System - Elements and organization; Operating System Overview - Objectives andFunctions - Evolution of Operating System; Operating System Structures – Operating System
Services - User Operating System Interface - System Calls – System Programs - Design andImplementation - Structuring methods

UNIT II PROCESS MANAGEMENT   9

Processes - Process Concept - Process Scheduling - Operations on Processes - Inter-process Communication; CPU Scheduling - Scheduling criteria - Scheduling algorithms: Threads - Multithread Models – Threading issues; Process Synchronization - The Critical-Section problem - Synchronization hardware – Semaphores – Mutex - Classical problems of synchronization - Monitors; Deadlock - Methods for handling deadlocks, Deadlock prevention, Deadlock avoidance, Deadlock detection, Recovery from deadlock.

UNIT III MEMORY MANAGEMENT   9

Main Memory - Swapping - Contiguous Memory Allocation – Paging - Structure of the Page Table
- Segmentation, Segmentation with paging; Virtual Memory - Demand Paging – Copy on Write -Page Replacement - Allocation of Frames –Thrashing.

UNIT IV STORAGE MANAGEMENT   9

Mass Storage system – Disk Structure - Disk Scheduling and Management; File-System Interface -File concept - Access methods - Directory Structure - Directory organization - File system mounting
- File Sharing and Protection; File System Implementation - File System Structure - Directoryimplementation - Allocation Methods - Free Space Management; I/O Systems – I/O Hardware,
Application I/O interface, Kernel I/O subsystem.

UNIT V VIRTUAL MACHINES AND MOBILE OS   9

Virtual Machines – History, Benefits and Features, Building Blocks, Types of Virtual Machines andtheir Implementations, Virtualization and Operating-System Components; Mobile OS - iOS and Android.

NUMBER OF THEORY PERIODS :   45

PRACTICAL EXERCISES:
1. Installation of windows operating system
2. Illustrate UNIX commands and Shell Programming
3. Process Management using System Calls : Fork, Exit, Getpid, Wait, Close
4. Write C programs to implement the various CPU Scheduling Algorithms
5. Illustrate the inter process communication strategy
6. Implement mutual exclusion by Semaphore
7. Write C programs to avoid Deadlock using Banker's Algorithm
8. Write a C program to Implement Deadlock Detection Algorithm
9. Write C program to implement Threading
10. Implement the paging Technique using C program
11. Write C programs to implement the following Memory Allocation Methods
      a.   First Fit b. Worst Fit c. Best Fit
12. Write C programs to implement the various Page Replacement Algorithms
13. Write C programs to Implement the various File Organization Techniques
14. Implement the following File Allocation Strategies using C programs a. Sequential b. Indexed c.Linked
15. Write C programs for the implementation of various disk scheduling algorithms

NUMBER OF PRACTICAL PERIODS :	30
TOTAL NUMBER OF PERIODS :	75

COURSE OUTCOMES:

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

CO1   Analyze various scheduling algorithms and process synchronization.

CO2   Explain deadlock prevention and avoidance algorithms

CO3   Compare and contrast various memory management schemes

CO4   Explain the functionality of file systems, I/O systems, and Virtualization

CO5   Compare iOS and Android Operating Systems

TEXTBOOKS:

1. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, "Operating System Concepts", 10thEdition, John Wiley and Sons Inc., 2018.

2. Andrew S Tanenbaum, "Modern Operating Systems", Pearson, 5th Edition, 2022 New Delhi.

REFERENCES:

1. Ramaz Elmasri, A. Gil Carrick, David Levine, "Operating Systems – A Spiral Approach", TataMcGraw Hill Edition, 2010.
2. William Stallings, "Operating Systems: Internals and Design Principles", 7 th Edition, PrenticeHall, 2018.

Achyut S.Godbole, Atul Kahate, "Operating Systems", McGraw Hill Education, 2016.

Course Code	INTRODUCTION TO ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING	L	T	P	C
IT3403		3	0	2	4

COURSE OBJECTIVES:
1. Study about uninformed and Heuristic search techniques
2. Learn Knowledge Representation and reasoning under uncertainty
3. Introduce Machine Learning and supervised learning algorithms
4. Study about ensembling and unsupervised learning algorithms
5. Learn the basics of deep learning using neural networks

UNIT I PROBLEM SOLVING   9

Introduction to AI - AI Applications - Problem solving agents – search algorithms – uninformed search strategies – Heuristic search strategies – Local search and optimization problems – adversarialsearch – constraint satisfaction problems (CSP).

UNIT II KNOWLEDGE REPRESENTATION   9

Knowledge, reasoning, and planning. Logical Agents, Knowledge-Based Agents, Propositional Logic: A Very Simple Logic Propositional Theorem Proving, Effective Propositional Model Checking, Agents Based on Propositional Logic, First-Order Logic, Representation Revisited, Syntax and Semantics of First-Order Logic, Using First-Order Logic, Knowledge Engineering in First-Order Logic. **PROBABILISTIC REASONING:** Acting under uncertainty – Bayesian inference – naïve bayes models. Probabilistic reasoning, Bayesian networks – exact inference in BN
– approximate inference in BN – causal networks

UNIT III SUPERVISED LEARNING   9

Introduction- Machine Learning, Designing a Learning Model – Examples of Machine Learning Applications- Supervised Learning, Learning a Class from Examples, Vapnik-Chervonenkis (VC) Dimension, PAC Learning, Noise Learning Multiple Classes, Regression , Model Selection and Generalization, Dimensions of a Supervised Machine Learning Algorithm– Linear Regression Models: Least squares, single & multiple variables, Bayesian linear regression, gradient descent, Linear Classification Models: Discriminant function – Probabilistic discriminative model - Logistic
regression, Probabilistic generative model – Naive Bayes, Maximum margin classifier – Support vector machine, Decision Tree, Random forests.

UNIT IV ENSEMBLE TECHNIQUES AND UNSUPERVISED LEARNING   9

Combining multiple learners: Model combination schemes, Voting, Ensemble Learning - bagging, boosting, stacking, Unsupervised learning: K-means, Instance Based Learning: KNN, Gaussian mixture models and Expectation maximization. k-Nearest Neighbour Estimator, Generalization to Multivariate Data, Nonparametric Classification, Condensed Nearest Neighbour. Multiple Classes,
Discrimination by Regression, Support Vector Machines, Optimal Separating Hyperplane, The Non-separable Case: Soft Margin Hyperplane, Kernel Functions, Support Vector Machines for Regression

UNIT V NEURAL NETWORKS   9

Perceptron - Multilayer perceptron, activation functions, network training – gradient descent optimization – stochastic gradient descent, error backpropagation, from shallow networks to deep networks –Unit saturation (aka the vanishing gradient problem) – ReLU, hyperparameter tuning,
batch normalization, regularization, dropout.

NUMBER OF THEORY PERIODS :   45

PRACTICAL EXERCISES:

1. Implementation of Uninformed search algorithms (BFS, DFS)
2. Implementation of Informed search algorithms (A*, memory-bounded A*)
3. Implement naïve Bayes models
4. Implement Bayesian Networks
5. Build Regression models
6. Build decision trees and random forests
7. Build SVM models
8. Implement ensembling techniques
9. Implement clustering algorithms
10. Implement EM for Bayesian networks
11. Build simple NN models
12. Build deep learning NN models

NUMBER OF PRACTICAL PERIODS :	30
TOTAL NUMBER OF PERIODS :	75

COURSE OUTCOMES:

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

CO1   Use appropriate search algorithms for problem-solving
CO2   Knowledge representation and apply reasoning under uncertainty
CO3   Build supervised learning models
CO4   Build ensembling and unsupervised models
CO5   Build deep-learning neural network models

TEXTBOOKS:

1. Stuart Russell and Peter Norvig, "Artificial Intelligence – A Modern Approach", Fourth Edition,Pearson Education, 2021.

Ethem Alpaydin, "Introduction to Machine Learning", MIT Press, Fourth Edition, 2020

REFERENCES:

1. Dan W. Patterson, "Introduction to Artificial Intelligence and Expert Systems", Pearson Education,2007
2. Kevin Night, Elaine Rich, and Nair B., "Artificial Intelligence", McGraw Hill, 2008
3. Deepak Khemani, "Artificial Intelligence", Tata McGraw Hill Education, 2013(http://nptel.ac.in/)
4. Christopher M. Bishop, "Pattern Recognition and Machine Learning", Springer, 2006.
5. Charu C. Aggarwal, "Data Classification Algorithms and Applications", CRC Press, 2014
6. Mehryar Mohri, Afshin Rostamizadeh, Ameet Talwalkar, "Foundations of Machine Learning",MIT Press, 2012.
7. Ian Goodfellow, Yoshua Bengio, Aaron Courville, "Deep Learning", MIT Press, 2016.

S Sridhar, M Vijayalakshmi, Machine Learning, First Edition, Oxford University Press

Course Code	INTRODUCTION TO CYBERSECURITY	L	T	P	C
IT3404		**3**	**0**	**2**	**4**

COURSE OBJECTIVES:

1. To develop a comprehensive understanding of fundamental cybersecurity concepts, including cyber threats, preventive measures, and cyber security principles.
2. To explore networking basics, focusing on concepts, protocols, andarchitectures crucial to understanding cyber security measures.
3. To acquire knowledge on core security principles, including risk management practices, to build a solid foundation for implementing effective security measures.
4. To understand the principles of cryptography and encryption, exploring their role in securing data and communication channels in cybersecurity.
5. To develop the skills to create, implement, and enforce security policies and procedures, fostering a security-aware culture and mitigating cyber threats through effective governance.

UNIT I INTRODUCTION TO CYBERSECURITY CONCEPTS   9

Introduction to Cybersecurity Fundamentals - Cyber Threat Landscape -Cybersecurity Frameworks and Standards - Security Architecture and Models - Incident Response and Cybersecurity Incident Handling - Security Awareness and Training - Legal and Ethical Aspects of Cybersecurity - Emerging Trends in Cybersecurity.

UNIT II FUNDAMENTALS OF NETWORKING   9

Introduction to Networking Concepts - OSI Model Overview - TCP/IP Protocol Suite - Network Devices and Components - IP Addressing and Subnetting - Routing and Switching Basics - Wireless Networking Fundamentals - Network Security Principles.

UNIT III SECURITY FUNDAMENTALS   9

Introduction to Security Principles - Access Control and Authentication - Security Risk Management - Security Policies and Procedures - Security Incident Response - Security Awareness Training - Vulnerability Assessment and Management - Physical Security Considerations.

UNIT IV CRYPTOGRAPHY AND ENCRYPTION   9

Introduction to cryptography - symmetric encryption algorithms - asymmetric encryption and public key infrastructure (PKI) - hash functions and message digests - digital signatures - cryptographic key management - transport layer security (TLS) and secure sockets layer (SSL) - cryptography in blockchain technology.

UNIT V SECURITY POLICIES AND PROCEDURES   9

Introduction to security policies and procedures - policy development and implementation - access control policies - data classification and handling policies - incident response plans - security awareness training programs - compliance and regulatory policies - security auditing and monitoring procedures.

NUMBER OF THEORY PERIODS :   40

List of Exercises :

1. Create a presentation outlining a hypothetical cyber threat scenario, includingthe potential risks, vulnerabilities, and recommended preventive measures.
2. Simulate a network troubleshooting scenario, identifying and resolving common networking issues, and providing a detailed analysis of the process.

3. Develop a scenario involving access control challenges within an organization, proposing solutions and policies to enhance security.
4. Develop a program or script to simulate the verification of digital signatures, emphasizing the importance of this cryptographic technique in ensuring data integrity.
5. Simulate a compliance audit scenario, identifying potential compliance issues and proposing measures to address and rectify them.
6. Create a simulated phishing campaign to test and enhance employees' awareness of phishing threats. Analyze the results and develop recommendations for ongoing awareness training.
7. Conduct a tabletop exercise simulating a cybersecurity incident. Develop a detailed response plan and practice its execution, involving relevant stakeholders and evaluating the effectiveness of the plan.
8. Provide a set of digital evidence related to a hypothetical cyber incident. Task students with conducting a forensic analysis, identifying the source of theincident, and preparing a detailed forensic report.
9. Evaluate the security of Internet of Things (IoT) devices within a given environment. Identify potential vulnerabilities and propose security measures to protect against IoT-related threats.
10. Organize a red team vs. blue team simulation, where one group simulates attackers (red team) and the other defends (blue team). Evaluate the effectiveness of defense mechanisms and identify areas for improvement.

NUMBER OF PRACTICAL PERIODS :	30
TOTAL NUMBER OF PERIODS :	70

COURSE OUTCOMES:
On completion of this course, the students will be able to:
CO1   Gain a clear understanding of fundamental cybersecurity principles, exploring the core concepts that form the basis of cyber defense
CO2   Understand the application of cybersecurity policies and risk management practices, emphasizing their importance in maintaining a secure digital environment
CO3   Learn about the basic tools and techniques for investigating various cybercrimes, providing a foundational knowledge base for cybercrime detection and resolution
CO4   Apply security fundamentals to digital environments, incorporating knowledge of risk management, access control, and other core principles to enhance overall security
CO5   Demonstrate proficiency in cyber forensics techniques and methodologies, integrating knowledge from security policies and procedures to enhance investigative capabilities

Assignments:
1. Analyze a real-world cybersecurity case study, identifying the threats, preventive measures, and the application of fundamental cybersecurity principles.
2. Develop a comprehensive proposal for a secure network design, considering protocols, architectures, and security measures to address potential threats.

3. Create a risk management plan for a fictional organization, outlining potential risks, mitigation strategies, and a framework for implementing security policies.

4. Evaluate the implementation of cryptographic techniques in a given system, assessing their effectiveness in securing data and communication.

5. Draft an incident response plan for a specific type of cybersecurity incident, detailing the steps to be taken during detection, analysis, and resolution.

6. Design and implement a comprehensive security awareness campaign for anorganization. Include strategies for educating employees on cybersecurity best practices, phishing awareness, and the importance of maintaining a security-conscious culture.

7. Conduct a simulated penetration test on a network or system, identifying vulnerabilities and providing a detailed report with recommendations for remediation.

8. Evaluate an organization's compliance with relevant cybersecurity regulations (e.g., GDPR, HIPAA). Develop a report outlining areas of compliance and suggesting improvements if necessary.

9. Perform a threat hunting exercise using security tools and techniques to proactively identify potential threats within a network. Provide a report detailing the findings and proposed actions.

10. Analyze and review the cybersecurity policies of a chosen organization. Identify strengths, weaknesses, and areas for improvement. Develop a report with recommendations for enhancing policy effectiveness.

TEXTBOOKS LINKS:
Unit 1: Introduction to Cybersecurity by jeetendra pande
https://uou.ac.in/sites/default/files/slm/Introduction-cyber-security.pdf **Unit 2: Fundamentals of Networking:**
https://www.cisco.com/c/dam/global/fi_fi/assets/docs/SMB_University_120307_Networking_Fundamentals.pdf
Unit 3: Security Fundamentals:
https://training.apnic.net/wp-content/uploads/sites/2/2016/11/eSEC01_NetSec.pdf
Unit 4: **Cryptography and Encryption:**
https://www.cs.umd.edu/~waa/414-F11/IntroToCrypto.pdfUnit 5: **Security Policies and**
Procedures:
https://oklahoma.gov/content/dam/ok/en/omes/documents/InfoSecPPG.pdf

REFERENCES:
1. Diogenes, Y., & Ozkaya, E. (2018). Cybersecurity - Attack and defense strategies: Infrastructure security with red team and blue team tactics. Packt Publishing.
2. Schneier, B. (2017). Applied cryptography: Protocols, algorithms and source code in C. John Wiley & Sons.
3. Erickson, J. (2008). Hacking: The art of exploitation (2nd ed.). No Starch Press.

Course Code	SOFTWARE ENGINEERING	L	T	P	C
IT3405		**3**	**0**	**2**	**4**

COURSE OBJECTIVES:

1. To understand Software Engineering Lifecycle Models

2. To Perform software requirements analysis

3. To gain knowledge of the System Analysis and Design concepts using UML.
4. To understand software testing and maintenance approaches
5. To work on project management scheduling using DevOps

Course Description

This course equips students with a strong foundation in software engineering principles and practices, preparing them for successful careers in the dynamic and evolving field of software development.

UNIT I SOFTWARE PROCESS AND AGILE DEVELOPMENT   9

Introduction to Software Engineering – Software Process – Perspective Models: Waterfall, Iterative & Incremental, V- and Spiral models – Specialized Process Models: Agile models, Component-Based Development (CBD), Rapid Application Development (RAD), and Formal Methods.

UNIT II   REQUIREMENTS ANALYSIS AND SPECIFICATION   9

Requirement Analysis and Specification – Requirements Gathering and Analysis – Functional and Non-Functional Requirements – Software Requirement Specification (SRS) – Formal System Specification: Finite State Machines, Petri Nets – Modern Requirement Modelling Techniques – Introduction to UML: Use Case Model, Class Diagrams – Agile Requirements Techniques: User Stories, Epics, Acceptance Criteria.

UNIT III   SOFTWARE DESIGN   9

Software Design Process and Concepts – Design Principles: Coupling, Cohesion, Functional Independence – Design Patterns: Model-View-Controller, Publish-Subscribe, Adapter, Command, Strategy, Observer, Proxy, Façade – Architectural Styles: Layered, Client-Server, Tiered, Pipe and Filter – User Interface Design – Modern Design Practices: Microservices Architecture, RESTful Services.

UNIT IV SOFTWARE TESTING AND MAINTENANCE   9

Testing Techniques: Unit Testing, Black Box Testing, White Box Testing, Integration and System Testing, Regression Testing – Debugging and Program Analysis – Advanced Testing Techniques: Symbolic Execution, Model Checking – Release Management: Continuous Integration, Continuous Deployment, DevOps Practices.

UNIT V PROJECT MANAGEMENT   9

Software Project Management Fundamentals – Software Configuration Management – Project Scheduling and Agile Project Management – DevOps: Motivation, Cloud as a Platform, Operations – Deployment Pipeline: Architecture, Building, Testing, Deployment – Tools for DevOps: Jenkins, Docker, Kubernetes.

NUMBER OF THEORY PERIODS :   45

Course Format

Lectures and discussions, a running Case Study throughout the course, Guest lectures by industry Experts, Group discussions and presentations, Online resources and tutorials

Assessments & Grading

Quizzes / Assignments, Project, 3 Internal Assessments, Final Examination

COURSE OUTCOMES:

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

CO1: Understand and apply various software development lifecycle models, with a focus on agile methodologies, to effectively manage software projects and adapt to changing requirements.
CO2: Perform comprehensive software requirements analysis and develop clear specifications using formal methods and techniques, ensuring the alignment of software solutions with stakeholder needs.
CO3: Employ appropriate modelling techniques for system analysis and design, incorporating architectural styles and design patterns to architect scalable and maintainable software systems.
CO4: Design and implement robust testing strategies to validate system functionality, reliability, and performance, ensuring the effective maintenance and evolution of software systems over time.
CO5: Evaluate project management approaches, including DevOps principles, and develop effective project plans and schedules to optimize project outcomes and manage software development projects efficiently.

TEXT BOOKS:

4. Roger Pressman, Bruce Maxim. "Software Engineering: A Practitioners Approach", 9th Edition, McGraw-Hill Education, 2019.
5. Marish Virzah, "Essential Scrum: A Practical Guide to the Most Popular Agile Process", Addison-Wesley, 2024.
6. Eric Freeman, Elisabeth Robson, "Head First Design Patterns", Second Edition, O'Reilly Media, 2021.
7. Gene Kim, Jez Humble, Patrick Debois, John Willis, Nicole Forsgren, "The DevOps Handbook", Second Edition, IT Revolution Press, 2021.

REFERENCES:

4. Sam Newman, "Building Microservices: Designing Fine-Grained Systems", Second Edition, O'Reilly Media, 2021.
5. Rafal Leszko, "Continuous Delivery with Docker and Jenkins", 3rd Edition, Packt Publishing, 2022.
6. Catherine Nelson, "Software Engineering for Data Scientists: From Notebooks to Scalable Systems", O'Reilly Media, 2024.

Course Code	Employability Enhancement Skills – III	L	T	P	C
ES3201	Leadership and Project Management Skills	0	0	2	1

COURSE OBJECTIVES:

The main objectives of this course are to:

- To understand leadership within the context of project management.
- To differentiate between leadership and management roles.
- To learn project initiation processes and setting SMART objectives.
- To build high-performing teams through motivation, empowerment, and effective communication.
- To develop skills in project planning, estimation, resource allocation, risk management, and scheduling..

UNIT I   **9**

Understanding Leadership - Introduction to Project Management - Leadership vs. Management - Project Initiation - Setting SMART Objectives.

UNIT II   **9**

Building High-Performing Teams - Motivation Theories - Empowering Team Members - Leadership Communication - Handling Team Conflicts.

UNIT III   **9**

Work Breakdown Structure (WBS) - Estimation Techniques - Gantt Charts and Network Diagrams - Resource Allocation - Risk Management.

UNIT IV   **9**

Leading Project Teams - Monitoring and Controlling Progress - Change Management - Quality Management - Stakeholder Communication.

UNIT V   **9**

Project Closure Activities - Lessons Learned - Celebrating Success - Transition Planning.

COURSE OUTCOMES:

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

CO1: Apply leadership principles to project management scenarios.
CO2: Distinguish between leadership and management functions in project environments.
CO3: Initiate projects effectively by setting SMART objectives.
CO4: Foster high-performing teams through motivation, empowerment, and conflict resolution.
CO5: Proficiently plan, schedule, and manage project activities, resources, risks, and stakeholder communications.

TEXT BOOKS:

1. Peter G. Northouse. "Leadership: Theory and Practice", SAGE Publications, 2021
2. Patrick Lencioni. "The Five Dysfunctions of a Team: A Leadership Fable", Jossey-Bass, 2011
3. Robert K. Wysocki. "Effective Project Management: Traditional, Agile, Extreme", Wiley, 2019
4. Clifford F. Gray and Erik W. Larson. "Project Management: The Managerial Process", McGraw-Hill Education, 2017
5. Harold Kerzner. "Project Management Case Studies", Wiley, 2008.

REFERENCES:

1. Harold Kerzner. "Project Management: A Systems Approach to Planning, Scheduling, and Controlling", 10th edition, Wiley, 2009
2. Gregory P. Shea and Cassie A. Solomon. "Leading Successful Change: 8 Keys to Making Change Work", Gildan Media, LLC, 2013.

TOTAL: 30 PERIODS

Course Code	PRINCIPLES OF COMMUNICATION SYSTEMS	L	T	P	C
IT3501		3	0	2	4

COURSE OBJECTIVES:

1 To understand the various devices used in Analog Communication
2 To comprehend the impact of interference in signaling devices
3 To learn the various issues in communication systems
4 To understand the digital transmission systems.
5 Imprehend the impact of interference in signaling devices

UNIT 1: INTRODUCTION TO DIGITAL COMMUNICATION   8

Introduction, Shannon limit for information capacity, digital amplitude modulation, frequency shift keying, FSK bit rate and baud, FSK transmitter, BW consideration of FSK, FSK receiver, phase shift keying –binary phase shift keying QPSK, Quadrature Amplitude modulation, bandwidth efficiency, carrier recovery types- squaring loop, Costas loop, DPSK.

UNIT - II Digital Transmission   11

Introduction, Pulse modulation, PCM sampling, sampling rate, signal to quantization noise rate, companding analog and digital percentage error, delta modulation, adaptive delta modulation, differential pulse code modulation, pulse transmission types-Intersymbol interference, eye patterns.

UNIT - III Amplitude Modulation Systems   9

Review of Spectral Characteristics of Periodic and Non-periodic signals; Generation and Demodulation of AM, DSBSC, SSB and VSB Signals; Comparison of Amplitude Modulation Systems

UNIT - Angle Modulation Systems   9

Frequency Translation; Non – Linear Distortion; Phase and Frequency Modulation; Single tone, Narrow Band and Wideband FM; Transmission Bandwidth; Generation and Demodulation of FM Signal, FDM and OFDM.

UNIT - V Wireless Networks   8

Introduction to wireless communication systems and networks; Wireless technologies: Cellular wireless networks and systems principles. Antennas and radio propagation. Signal encoding and modulation techniques. Spread spectrum. Coding and error control. Wireless Networking: Multiple access techniques. Mobile IP and WAP. Wireless systems and standards. Wireless LANs: Wireless LAN technology. Wireless standard (IEEE 802.11 etc.). Ad-hoc Networks. Bluetooth.

TOTAL PERIODS :   45

LIST OF EXPERIMENTS:

1 **Study and analyze the** functional modules of a **Amplitude Modulation Systems**

2 **Study and analyze the** functional flow of FSK,DPSK, QPSK generation schemes
3 Generation and Detection of Pulse Code Modulation for A.C and D.C signals

4	Generation and Detection of Pulse Code Modulation for D.C signals
5	Use relevant tool for simulating Wireless Network.
6	Test the various Input/output devices of mobile phone
7	Develop a mobile application for wireless technology using any wizards such as available on www.appypie.com or any other

COURSE OUTCOMES:

After the completion of the course the student will be able to

CO1	Demonstrate the knowledge of fundamental elements and concepts related to Communication System.
CO2	Design and construct devices used in Communication Systems
CO3	Address the challenges imposed on different types of Communication Systems.
CO4	Use and apply important methods in communication systems to support both analog and digital communication.
CO5	Provide solutions to digital communication by using different modulation techniques.

TEXTBOOKS:

1. Analog and Digital Communications, Sudakshina Kundu, Pearson Education 2010.

REFERENCES:

1. Herbert Taub & Donald L Schilling, Principles of Communication Systems, Third Edition, Tata McGraw Hill, 2013.
2. W. Stallings, "Wireless Communications and Networks", Pearson Education, 2002.
3.J Schiller, "Mobile Communications", Addison Wesley, 2000.

4.Bruce Carlson, Communication Systems, Third Edition, McGraw Hill.

Course Code	EMBEDDED PROGRAMMING	L	T	P	C
IT3502		3	0	2	4

COURSE OBJECTIVES:
1. To expose the addressing modes & instruction set of 8085.
2. To study the architecture and embedded C programming and its salient features forembedded systems
3. To learn about various IOT-related protocols.
4. To develop IoT infrastructure for different applications.
5. To involve Discussions/ Practice/Exercise in revising & familiarizing the conceptsacquired over the 5 Units of the subject for improved employability skills.

UNIT 1: 8086 MICROPROCESSOR 9

8086 Architecture – Memory interfacing–I/O ports and data transfer concepts – Timing Diagram –Interrupt structure- DMA - Instruction set and addressing modes - Interfacing: 8255 PPI, 8279
Keyboard display controller, and 8254 Timer/Counter.

UNIT - II 8-BIT EMBEDDED PROCESSOR 9

8-Bit Microcontroller – Architecture – Instruction Set and Programming –Pipelining - ProgrammingParallel Ports – Timers and Serial Port – Interrupt Handling - ARM Architecture.

UNIT - III EMBEDDED C PROGRAMMING 9

Memory And I/O Devices Interfacing – Programming Embedded Systems in C – Need for RTOS –Multiple Tasks and Processes – Context Switching – Priority Based Scheduling Policies.

UNIT - IV IoT ARCHITECTURE AND PROTOCOLS 9

Definition & Characteristics of IoT - Challenges and Issues - Physical Design of IoT - Logical Designof IoT - IoT Enabling Technologies – Domain Specific IoTs – IoT and M2M - IoT Communication
Models and APIs – IoT Protocols – LoRaWAN, 6LoWPAN, CoAP, MQTT.

UNIT - V IoT SYSTEM DESIGN 9

IoT Systems logic design using Python–packages for IoT- IoT Physical device & Endpoints-Basicbuilding blocks of an IoT Device – Single board computer.

TOTAL PERIODS : 45
PRACTICAL EXERCISE 30 PERIODS

Laboratory exercise: Use any Embedded processor/IDE/open source platform
to givehands-on training on basic concepts of embedded system design:

1 Write 8051 Assembly Language experiments using simulator.
2 Data transfer between registers and memory.
3 Write Basic and arithmetic Programs Using Embedded C.
4 Programming with IDE – ARM
5 Introduction to Raspberry PI platform and python programming
6 Interfacing sensors with Raspberry PI
7 Design IoT applications using STM32 /Jetson Nano/open platform.

Seminar: Students are able to give the seminar on recent trends in EmbeddedProcessor/IoT/Applications of IoT and other Programming tools etc.
Mini Project: Applications of IoT & Embedded System.

COURSE OUTCOMES:

After the completion of the course the student will be able to

CO1   Understand the addressing modes & instruction set of 8085.
CO2   Deliver insight into architecture and embedded C programming and its salient features for embedded systems.
CO3   Analyze various protocols for IoT.
CO4   Design an IoT based system for any applications.
CO5   Improved Employability and entrepreneurship capacity due to knowledge upgradation on recent trends in embedded programming skills.

TOTAL: 75 PERIODS

TEXTBOOKS:

1. Marilyn Wolf, Computers as Components – Principles of Embedded Computing System Design, Third Edition, Morgan Kaufmann, 2012.
2. Muhammed Ali Mazidi, Janice Gillispie Mazidi, Rolin D. McKinlay, "The 8051 Microcontroller and Embedded Systems", Pearson Education, Second Edition, 2014
3. Robert Barton, Patrick Grossetete, David Hanes, Jerome Henry, Gonzalo Salgueiro, "IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things", CISCO Press, 2017

REFERENCES:

1. Ramesh S. Gaonkar, 'Microprocessor Architecture Programming and Application',Penram International (P)ltd., Mumbai, 6th Education, 2013.
2. Michael J. Pont, "Embedded C", Pearson Education, 2007.
3. Arshdeep Bahga, Vijay Madisetti, "Internet of Things – A hands-on approach", Universities Press,2015
4. Mayur Ramgir, Internet – of – Things, Architecture, Implementation and Security, First Edition, Pearson Education, 2020.

Course Code	DATA ANALYTICS AND VISUALIZATION	L	T	P	C
IT3503		**3**	**0**	**2**	**4**

COURSE OBJECTIVES:
1. To understand the data science fundamentals and process.
2. To learn to describe the relationship between data.
3. To utilize present and interpret data using Python libraries for Data Wrangling and data visualization.
4. To study the basic inferential statistics, sampling distribution and processes in data analytics
5. To understand the data analytics techniques and apply descriptive data analytics techniques

UNIT I INTRODUCTION   **6**

Data Science: Benefits and uses – facets of data - Data Science Process: Overview – Defining research goals – Retrieving data – Data preparation - Exploratory Data analysis – build the model– presenting findings and building applications - Data Mining - Data Warehousing – Basic Statistical descriptions of Data.

UNIT II DESCRIBING DATA   **6**

Basics of Numpy arrays –aggregations –computations on arrays –comparisons, masks, boolean logic – fancy indexing – structured arrays – Data manipulation with Pandas – data in Types of Data - Types of Variables -Describing Data with Tables and Graphs –Describing Data with Averages - Describing Variability - Normal Distributions and Standard (z) Scores.
DESCRIBING RELATIONSHIPS: Correlation –Scatter plots –correlation coefficient for quantitative data –computational formula for correlation coefficient – Regression –regression line –least squares regression line – Standard error of estimate – interpretation of r2 –multiple regression equations –regression towards the mean

UNIT III PYTHON LIBRARIES FOR DATA WRANGLING   **6**

Indexing and selection – operating on data – missing data – Hierarchical indexing – combining datasets – aggregation and grouping – pivot tables
DATA VISUALIZATION: Importing Matplotlib – Line plots – Scatter plots – visualizing errors – density and contour plots – Histograms – legends – colors – subplots – text and annotation – customization – three-dimensional plotting - Geographic Data with Basemap - Visualization with Seaborn.

UNIT IV DESCRIPTIVE ANALYTICS AND INFERENTIAL STATISTICS   **6**

DESCRIPTIVE ANALYTICS - Frequency distributions – Outliers –interpreting distributions – graphs – averages - describing variability – interquartile range – variability for qualitative and ranked data - Normal distributions – z scores –correlation – scatter plots – regression – regression line – least squares regression line – standard error of estimate – interpretation of r2 – multiple regression equations – regression toward the mean.

INFERENTIAL STATISTICS - populations – samples – random sampling – Sampling distribution- standard error of the mean - Hypothesis testing – z-test – z-test procedure – decision rule – calculations – decisions – interpretations - one-tailed and two-tailed tests – Estimation – point estimate – confidence interval – level of confidence – effect of sample size.

UNIT V ANALYSIS OF VARIANCE AND PREDICTIVE ANALYTICS   **6**

ANALYSIS OF VARIANCE - T-test for one sample – sampling distribution of t – t-test procedure – t-test for two independent samples – p-value – statistical significance – t-test for two related samples. F-test – ANOVA – Two-factor experiments – three f-tests – two-factor ANOVA – Introduction to chi-square tests.

PREDICTIVE ANALYTICS - Linear least squares – implementation – goodness of fit – testing a linear model – weighted resampling. Regression using Stats Models – multiple regression –

nonlinear relationships – logistic regression – estimating parameters – Time series analysis – moving averages – missing values – serial correlation – autocorrelation. Introduction to survival analysis.

TOTAL NUMBER OF PERIODS INCLUDING LAB: 30

SAMPLE LIST OF EXPERIMENTS   **NUMBER OF 30**
PRACTICAL PERIODS:

1. Download, install and explore the features of NumPy, SciPy, Jupyter, Statsmodels and Pandas packages, Scipy, Matplotlib, Pandas, statmodels, seaborn, plotly, bokeh
2. Working with Numpy arrays
3. Working with Pandas data frames
4. Reading data from text files, Excel and the web and exploring various commands for doing descriptive analytics on the Iris data set.
5. Use the diabetes data set from UCI and Pima Indians Diabetes data set for performing the following:
a. Univariate analysis: Frequency, Mean, Median, Mode, Variance, Standard Deviation, Skewness and Kurtosis.
b. Bivariate analysis: Linear and logistic regression modeling
c. Multiple Regression analysis
d. Also compare the results of the above analysis for the two data sets.
6. Apply and explore various plotting functions on UCI data sets.
a. Normal curves
6. Perform Z-test
8. Perform T-test
9. Perform ANOVA
10. Building and validating linear models
11. Building and validating logistic models
12. Time series analysis

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Define the data science process Understand different types of data descriptions for data science process

CO2   Gain knowledge on relationships between data and use the Python Libraries for Data Wrangling

CO3   Apply visualization Libraries in Python to interpret and explore data

CO4   Perform various statistical analyses to make statistical inferences and explain the end-to-end data analytics pipeline

CO5   Build, validate and communicate data analytical models for complex engineering problems

TEXTBOOKS:

1. David Cielen, Arno D. B. Meysman, and Mohamed Ali, "Introducing Data Science", Manning Publications, 2016.
2. Robert S. Witte and John S. Witte, "Statistics", Eleventh Edition, Wiley Publications, 2017. Padeepz App Padeepz App 69
3. Jake VanderPlas, "Python Data Science Handbook", O'Reilly, 2016.

REFERENCES:

1. Allen B. Downey, "Think Stats: Exploratory Data Analysis in Python", Green Tea Press,2014.
2. Peter Bruce, Andrew Bruce, and Peter Gedek, "Practical Statistics for Data Scientists", Second Edition, O'Reilly Publishers, 2020.
3. Charles R. Severance, "Python for Everybody: Exploring Data in Python 3", Shroff Publishers, 2017
4. Bradley Efron and Trevor Hastie, "Computer Age Statistical Inference", Cambridge University Press, 2016

Course Code	Employability Enhancement Skills - IV	L	T	P	C
ES3201		0	0	2	1

COURSE OBJECTIVES:

The main objectives of this course are to:

1. To give the exposure to solve the real time problems related to company test questions.
2. To train the students to provide solutions for the assigned tasks.

PRACTICAL EXERCISES: **30 PERIODS**

Questions from top notch industries – Programming, Quantitative Aptitude – Verbal Ability – Numerical Ability – Reasoning Ability – Mock Test.

COURSE OUTCOMES:

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

1. **CO1: T**o solve the real time programming problems related to company test questions.
2. **CO2:** To solve Quantitative Aptitude – Verbal Ability – Numerical Ability – Reasoning Ability Questions.

Course Code	MANAGERIAL ECONOMICS & FINANCIAL	L	T	P	C
HS3601	ACCOUNTING	3	0	0	3

COURSE OBJECTIVES:

1. Understanding the concept of Engineering Economics.

2. Gaining knowledge in macroeconomics to give the students a better understanding.

3. understanding of various components of macroeconomics.

4. Understanding the different procedures of pricing.

5. Learn the various cost-related concepts in microeconomics.

UNIT I DEMAND & SUPPLY ANALYSIS 9

Managerial Economics - Relationship with other disciplines - Firms: Types, objectives, and goals - Managerial decisions - Decision analysis. Demand - Types of demand - Determinants of demand - Demand function – Demand elasticity - Demand forecasting - Supply - Determinants of supply - Supply function -Supply elasticity.

UNIT II PRODUCTION AND COST ANALYSIS 9

Production function - Returns to scale - Production optimization - Least cost input - Isoquants - Managerial uses of production function. Cost Concepts - Cost function - Determinants of cost - Short run and Long run cost curves - Cost Output Decision - Estimation of Cost.

UNIT III PRICING 9

Determinants of Price - Pricing under different objectives and different market structures - Price discrimination - Pricing methods in practice.

UNIT IV FINANCIAL ACCOUNTING (ELEMENTARY TREATMENT) 9

Balance sheet and related concepts - Profit & Loss Statement and related concepts - - Financial Ratio Analysis - Cash flow analysis - Funds flow analysis - Comparative financial statements - Analysis & Interpretation of financial statements.

UNIT V CAPITAL BUDGETING (ELEMENTARY TREATMENT) 9

Investments - Risks and return evaluation of investment decision - Average rate of return - Payback Period - Net Present Value - Internal rate of return.

TOTAL NUMBER OF PERIODS : 45

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1 Students will acquire the skills to apply the basics of economics and cost analysis to engineering and take economically sound decisions

CO2 Evaluate the economic theories, cost concepts and pricing policies

CO3 Understand the market structures and integration concepts

CO4 Understand the measures of national income, the functions of banks and concepts of globalization

CO5 Apply the concepts of financial management for project appraisal

TEXTBOOKS:

1. Panneer Selvam, R, "Engineering Economics", Prentice Hall of India Ltd, New Delhi,2001.

2. Managerial Economics: Analysis, Problems and Cases - P. L. Mehta, Edition, 13. Publisher, Sultan Chand, 2007.

REFERENCES:

1. Chan S.Park, "Contemporary Engineering Economics", Prentice Hall of India, 2011.

2. Donald.G. Newman, Jerome.P.Lavelle, "Engineering Economics and analysis" Engg. Press, Texas, 2010.

3. Degarmo, E.P., Sullivan, W.G and Canada, J.R, "Engineering Economy", Macmillan, New York, 2011.

4. Zahid A khan: Engineering Economy, "Engineering Economy", Dorling Kindersley, 2012 5. Dr. S. N. Maheswari and Dr. S.K. Maheshwari: Financial Accounting,

		L	T	P	C
Course Code	**Cloud and Distributed Computing**				
IT3601		3	0	0	3

COURSE OBJECTIVES:
The objective of this course is to enable the students to

1. To Study about on-demand delivery of computations, storage, applications, and other IT resources through a cloud services platform
2. To Know the distributed system and cloud models.
3. To apply distributed computational model and understand the need for cloud computing.
4. To Learn about cloud recovery and storage.
5. To intimate knowledge about the internals of cloud computing and how the distributed systems concepts work inside clouds.

UNIT I INDRODUCTION TO DISTRIBUTED SYSTEM   9

Characterization of Distributed System: Introduction-Examples of distributed system, Trends in distributed system, focus on resource sharing, Challenges – System models – Inter Process Communication – Remote Invocation – Indirect Communication – Case study: World Wide Web.

UNIT II DISTRUIBUTED ALGORITHMS   10

Message Passing, Leader Election, Distributed Models, Causality and Logical Time, Global State & Snapshot and Distributed Mutual Exclusion- Token based approaches, Consensus & Agreement, Checkpointing & Rollback Recovery - Introduction classical distributed algorithms - Algorithm for Recording Global State and Snapshot - Time and Clock Synchronization in Cloud Data Center - Key Challenges.

UNIT III CLOUD VIRTUALIZATION   10

Features of Today's Cloud-Layered Cloud Architecture Design – NIST Cloud Computing Reference Architecture: public, private, hybrid. Service provider interfaces: SaaS, PaaS, IaaS – Virtualization Technology - Hardware Independence, Server Consolidation, Resource Replication, Operating System-Based Virtualization, Hardware-Based Virtualization, Virtualization management and considerations – Use case and Example.

UNIT IV CLOUD STORAGE AND RECOVERY   9

Fundamental cloud architectures - workload distribution, resource pooling, dynamic scalability, elastic resource capacity, service load balancing, cloud bursting, elastic disk provisioning, redundant storage, Case Study Example - Advanced Cloud Architectures, Hypervisor clustering, Load balanced virtual server instances, Non-Disruptive service relocation, Zero-downtime, Cloud balancing, Resource reservation, Dynamic failure detection and recovery, Bare-metal provisioning, Rapid provisioning, Storage workload management, Case Study Example.

UNIT V CLOUD PLATFORMS AND APPLICATIONS   9

Cloud platform: Microsoft Azure, Amazon Web Services, Google Cloud, IBM Cloud and CloudLinux–APIs - Microservices - Docker - Kubernetes - Applications and Use Cases.

TOTAL NUMBER OF PERIODS : 45

COURSE OUTCOMES:
On completion of this course, the students will be able to:
CO1   Explain the foundations of distributed systems.
CO2   Solve synchronization and state consistency problems.
CO3   Understand how distributed system concepts apply inside cloud.

CO4   Apply the distributed algorithms for effective scheduling.
CO5   Analyze the need for virtualization and apply it in a cloud environment.

TEXTBOOKS:

1. Rajiv Misra, Yashwant Singh Patel, Cloud and Distributed Computing: Algorithms and Systems, Wiley India.

2. Coulouris George, Dollimore Jean, Kindberg Tim, Blair Gordon, Distributed Systems Concepts and Design, Pearson Education, 5th Edidtion

REFERENCES:

Danielle Ruest, Nelson Ruest, Virtualization: A Beginner's Guide, McGraw-Hill Osborne Media, 2009.

2. Thomas Erl, Ricardo Puttini, Zaigham Mahmood, Cloud Computing: Concepts, Technology & Architecture, First Edition, Pearson Education 2013.

3. John W.Rittinghouse and James F.Ransome, Cloud Computing: Implementation, Management, and Security, CRC Press, 2010.

4. Michael J.Kavis, "Architecting the Cloud". Wiley India, 2014.

LIST of Lab sessions

1. Inter-process communication
2. Remote Method Invocation
3. Message passing and distributed mutual exclusion
4. Clock and Time synchronization
5. Virtual instances and server consolidation with fault-tolerance
6. Task scheduling and load balancing
7. Hypervisor clustering
8. Hosting microservices on cloud platforms
9. Dockerization and Kubernetes
10. Mini project

Course Code	MOBILE COMPUTING	L	T	P	C
IT3602		3	0	2	4

COURSE OBJECTIVES:

1. To understand the basic concepts of mobile computing.

2. To learn the basics of mobile telecommunication system .

3. To be familiar with the network layer protocols and Ad-Hoc networks.

4. To know the basis of transport and application layer protocols.

5. To gain knowledge about different mobile platforms and application development

UNIT I INTRODUCTION   **9**

Introduction to Mobile Computing – Applications of Mobile Computing- Generations of Mobile Communication Technologies- Multiplexing – Spread spectrum -MAC Protocols – SDMA- TDMA- FDMA- CDMA

UNIT II MOBILE TELECOMMUNICATION SYSTEM   **9**

Introduction to Cellular Systems - GSM – Services & Architecture – Protocols – Connection Establishment – Frequency Allocation – Routing – Mobility Management – Security – GPRS- UMTS – Architecture – Handover – Security.

UNIT III MOBILE NETWORK LAYER   **9**

Mobile IP – DHCP – AdHoc– Proactive protocol-DSDV, Reactive Routing Protocols – DSR,AODV , Hybrid routing –ZRP, Multicast Routing- ODMRP, Vehicular Ad Hoc networks (VANET) – MANET Vs VANET – Security

UNIT IV MOBILE TRANSPORT AND APPLICATION LAYER   **9**

Mobile TCP– WAP – Architecture – WDP – WTLS – WTP –WSP – WAE – WTA Architecture– WML.

UNIT VI MOBILE PLATFORMS AND APPLICATIONS   **9**

Mobile Device Operating Systems – Special Constraints & Requirements – Commercial Mobile Operating Systems – Software Development Kit: iOS, Android, BlackBerry, Windows Phone – MCommerce – Structure – Pros & Cons – Mobile Payment System – SecurityIssues

NUMBER OF THEORY PERIODS : 45

List of Practical Experiments :

1. Create a simple Android app that displays "Hello World" on the screen.

2. Create an app with a text field and a button. Display a toast message with the entered text when the button is clicked. Write test cases and generate test data for a given application.

3. Build a basic calculator app with operations like addition, subtraction, multiplication, and division. Conducting performance tests using a performance testing tool.

4. Create an app that fetches and displays the user's current location using GPS or network providers.

5. Case Study on IOS App Development

6. Create a simple iOS app that displays "Hello World" on the screen.

NUMBER OF PRACTICAL PERIODS : 30
TOTAL NUMBER OF PERIODS : 75

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Explain the basics of mobile telecommunication systems
CO2   Illustrate the generations of telecommunication systems in wireless networks
CO3   Determine the functionality of MAC, network layer and Identify a routing protocol for a given Ad hoc network
CO4   Explain the functionality of Transport and Application layers
CO5   Develop a mobile application using android/blackberry/Ios/Windows SDK

TEXTBOOKS:

1. Jochen Schiller, ―Mobile Communications‖, PHI, Second Edition, 2003.
2.Prasant Kumar Pattnaik, Rajib Mall, ―Fundamentals of Mobile Computing‖, PHI   Learning Pvt.Ltd, New Delhi – 2012

REFERENCES:

1. Dharma Prakash Agarval, Qing and An Zeng, "Introduction to Wireless and Mobile systems",Thomson Asia Pvt Ltd, 2005.
2. Uwe Hansmann, Lothar Merk, Martin S. Nicklons and Thomas Stober, ―Principles of
3. Mobile Computing‖, Springer, 2003.
4. William.C.Y.Lee,―Mobile Cellular Telecommunications-Analog and Digital Systems‖, Second Edition,TataMcGraw Hill Edition ,2006.
5. C.K.Toh, ―AdHoc Mobile Wireless Networks‖, First Edition, Pearson Education, 2002.
6. Android Developers : http://developer.android.com/index.html
7. Apple Developer : https://developer.apple.com/
8. Windows Phone DevCenter : http://developer.windowsphone.com
9. BlackBerry Developer : http://developer.blackberry.com

Semester VII

Course Code	PROFESSIONAL ETHICS AND UNIVERSAL HUMAN	L	T	P	C

HS3701	VALUES	3	0	0	3

COURSE OBJECTIVE:

To enable the students to create an awareness of Engineering Ethics and Human Values, to instill Moral and Social Values and Loyalty, and to appreciate the rights of others.

UNIT I HUMAN VALUES   **10**

Morals, values, and Ethics – Integrity – Work ethic – Service learning – Civic virtue – Respect for others – Living peacefully – Caring – Sharing – Honesty – Courage – Valuing time – Cooperation – Commitment – Empathy – Self-confidence – Character – Spirituality – Introduction to Yoga and meditation for professional excellence and stress management.

UNIT II ENGINEERING ETHICS   **9**

Senses of 'Engineering Ethics' – Variety of moral issues – Types of inquiry – Moral dilemmas – Moral Autonomy – Kohlberg's theory – Gilligan's theory – Consensus and Controversy – Models of professional roles - Theories about right action – Self-interest – Customs and Religion – Uses of Ethical Theories.

UNIT III ENGINEERING AS SOCIAL EXPERIMENTATION   **9**

Engineering as Experimentation – Engineers as Responsible Experimenters – Codes of Ethics – A Balanced Outlook on Law.

UNIT IV SAFETY, RESPONSIBILITIES, AND RIGHTS   **9**

Safety and Risk – Assessment of Safety and Risk – Risk-Benefit Analysis and Reducing Risk - Respect for Authority – Collective Bargaining – Confidentiality – Conflicts of Interest – Occupational Crime – Professional Rights – Employee Rights – Intellectual Property Rights (IPR) – Discrimination.

UNIT V GLOBAL ISSUES   **8**

Multinational Corporations – Environmental Ethics – Computer Ethics – Weapons Development – Engineers as Managers – Consulting Engineers – Engineers as Expert Witnesses and Advisors – Moral Leadership –Code of Conduct – Corporate Social Responsibility.

TOTAL: 45 PERIODS

COURSE OUTCOME:

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

CO1: Apply ethics in society, discuss the ethical issues related to engineering, and realize societal responsibilities and rights.

?

1. Mike W. Martin and Roland Schinzinger, "Ethics in Engineering", Tata McGraw Hill, New Delhi, 2003.

2. Govindarajan M, Natarajan S, Senthil Kumar V. S, "Engineering Ethics", Prentice Hall of India, New Delhi, 2004.

REFERENCES:

1. Charles B. Fleddermann, "Engineering Ethics", Pearson Prentice Hall, New Jersey, 2004.
2. Charles E. Harris, Michael S. Pritchard and Michael J. Rabins, "Engineering Ethics – Concepts and Cases", Cengage Learning, 2009.
3. John R Boatright, "Ethics and the Conduct of Business", Pearson Education, New Delhi, 2003
4. Edmund G Seebauer and Robert L Barry, "Fundamentals of Ethics for Scientists and Engineers", Oxford University Press, Oxford, 2001.
5. Laura P. Hartman and Joe Desjardins, "Business Ethics: Decision Making for Personal Integrity and Social Responsibility" Mc Graw Hill education, India Pvt. Ltd.,New Delhi, 2013.
6. World Community Service Centre, ' Value Education', Vethathiri publications, Erode, 2011.

Web sources:

1. www.onlineethics.org
2. www.nspe.org
3. www.globalethics.org
4. www.ethics.org

APPENDIX A: PROFESSIONAL ELECTIVE COURSES VERTICALS

Vertical I Full Stack Development	Vertical II Software Technologies	Vertical III Cyber Security	Vertical IV Artificial Intelligence and Data Science	Vertical V Cloud Computing	Vertical VI IoT Systems	Vertical VII High-End Computing
CSPE3XV11 Web Development 5.0	CSPE3X V21 Software Engineering	CSPE 3X V31 Cryptography & Information Security	CSPE3X V41 Data Analytics and Visualization	CSPE3X V51 Cloud Technologies	CSPE 3X V61 Foundations of Embedded IoT Systems	CSPE3X V71 Parallel Processing
CSPE 3X V12 App Development	CSPE 3X V22 Software Testing and Automation	CSPE 3X V32 Security and Privacy in Cloud	CSPE 3X V42 Mathematics of Machine Learning	CSPE 3X V52 Virtualization	CSPE3X V62 IoT Networks	CSPE 3X V72 Hi-Performance Computing
CSPE3X V13 UI and UX Design	CSPE 3X V23 Agile Methodologies	CSPE 3X V33 Web Application Security	CSPE 3X V43 Deep Learning	CSPE 3X V53 Cloud Architectures	CSPE3X V63 Secure Hardware and Embedded Devices	CSPE 3X V73 Pervasive Computing
CSPE3X V14 Cloud Services Management	CSPE 3X V24 Software User Interface Design & Analysis	CSPE 3X V34 Social Network Security	CSPE3X V44 Natural Language Processing (NLP)	CSPE 3X V54 Cloud Platform Programming	CSPE 3X V64 IoT Processors	CSPE 3X V74 Pico Computing
CSPE3X V15 DevOps	CSPE 3X V25 Software Architecture and Design Patterns	CSPE 3X V35 Digital Forensics and Malware Analysis	CSPE3X V45 Computer Vision and Image Processing	CSPE 3X V55 Cloud Services Management	CSPE 3X V65 Mobile Applications Development	CSPE 3X V75 Nano Computing
CSPE 3X V16 Web Frameworks	CSPE 3X V26 Software Quality Management	CSPE 3X V36 Ethical Hacking	CSPE3X V46 Reinforcement Learning	CSPE3X V56 Stream Processing	CSPE 3X V66 Industrial IoT & Healthcare Systems	CSPE 3X V76 Edge and Fog Computing
CSPE 3X V17 Middle Tier Technologies	CSPE 3X V27 Software Project Management	CSPE 3X V37 Cryptocurrency & Blockchain Technologies	CSPE 3X V47 Big Data Analytics	CSPE 3X V57 Fog and Edge Computing	CSPE3X V67 Smart Cities	CSPE3X V77 AI and Cloud Computing
CSPE 3X V18 Web Application Security	CSPE 3X V28 Human-Computer Interaction	CSPE V38 Security Auditing & Counter Hacking Techniques	CSPE 3X V48 Generative Adversarial Networks	CSPE3X V58 Blockchain Concepts and Technologies	CSPE3X V68 Advanced Intelligent Systems	CSPE3X V78 Quantum Computing

*Students are permitted to choose all the Professional Electives from a particular vertical or from different verticals. However, Students are restricted to select from not more than 2 verticals.

<div align="center">

Vertical I

FULL STACK DEVELOPMENT

</div>

Course Code	WEB DEVELOPMENT 5.0	L	T	P	C
CSPE3XV11		2	0	2	3

COURSE OBJECTIVES:

1. To define the evolution of web development from static websites to modern web applications
2. To explore the concepts of components, JSX, and state management.
3. To define Node.js and Express.js and their roles in server-side development.
4. To integrate databases with web applications.
5. To understand various deployment strategies for web applications.

UNIT I: INTRODUCTION TO MODERN WEB DEVELOPMENT   **6**

Evolution of Web Development- Historical overview of web development- Introduction to Web Development 5.0 and its characteristics- Modern Web Development Ecosystem- Overview of front-end and back-end technologies- Introduction to popular frameworks and libraries.

UNIT II: FRONT-END DEVELOPMENT WITH REACT.JS   **6**

Introduction to React.js- Overview of React.js and its key features- Understanding JSX syntax and component-based architecture- Components and State Management- Creating functional and class components- Managing state and props in React applications.

UNIT III: BACK-END DEVELOPMENT WITH NODE.JS AND EXPRESS.JS   **6**

Introduction to Node.js and Express.js- Overview of Node.js and its event-driven architecture- Introduction to Express.js as a web application framework- Building RESTful APIs- Understanding RESTful principles- Creating a simple RESTful API with Express.js

UNIT IV: DATABASE INTEGRATION AND AUTHENTICATION   **6**

Database Integration- Overview of database options (MongoDB, MySQL, etc.)- Integrating MongoDB with Express.js- User Authentication and Authorization- Introduction to authentication and authorization- Implementing user authentication using Passport.js

UNIT V: DEPLOYMENT AND PROJECT WORK   **6**

Deployment Strategies- Traditional Server Hosting- Cloud Hosting- Containerization- Container Orchestration- Serverless Architecture- Continuous Deployment- Blue-Green Deployment- Feature Toggles- Best Practices- Project Work Guidelines- Tips for effective project planning, collaboration, and troubleshooting

TOTAL NUMBER OF PERIODS INCLUDING LAB: 60

SAMPLE LIST OF EXPERIMENTS   **NUMBER OF PRACTICAL PERIODS: 30**

1. Installation and configuration of development tools.
2. Introduction to version control using Git.
3. Creating a basic HTML5 and CSS3 web page.
4. Introduction to responsive design principles.
5. Building a simple React application.
6. Implementing components and managing state.
7. Practical exercises on creating stateful and stateless components.
8. Implementing interactivity in React applications.
9. Creating a basic web server using Node.js.
10. Implementing asynchronous programming with callbacks and Promises.
11. Integration of MongoDB with Express.js.

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Understand and Apply Modern Web Development Concepts.

CO2   Create interactive and dynamic user interfaces with React.js.

CO3   Apply design patterns to new projects and re-factor existing code.

CO4   Implement secure user authentication and authorization using Passport.js.

CO5   Understand various deployment strategies and hosting platforms.

TEXTBOOKS:

1. "Learning React" by Alex Banks and Eve Porcello
2. "Node.js Design Patterns" by Mario Casciaro
3. "The DevOps Handbook" by Gene Kim, Jez Humble, Patrick Debois, and John Willis.

REFERENCES:

1. "Express in Action" by Evan Hahn
2. "MongoDB: The Definitive Guide" by Shannon Bradshaw, Eoin Brazil, Kristina Chodorow

Course Code	APP DEVELOPMENT	L	T	P	C
CSPE3XV12		**2**	**0**	**2**	**3**

COURSE OBJECTIVES:

1. To learn the development of native applications with basic GUI Components
2. To develop cross-platform applications with event handling
3. To develop applications with location and data storage capabilities
4. To develop web applications with database access

UNIT I FUNDAMENTALS OF MOBILE & WEB APPLICATION DEVELOPMENT   **6**

Basics of Web and Mobile application development, Native App, Hybrid App, Cross-platform App, What is Progressive Web App, Responsive Web design

UNIT II NATIVE APP DEVELOPMENT USING JAVA   **6**

Native Web App, Benefits of Native App, Scenarios to create Native App, Tools for creating Native App, Cons of Native App, Popular Native App Development Frameworks, Java & Kotlin for Android, Swift & Objective-C for iOS, Basics of React Native, Native Components, JSX, State, Props

UNIT III HYBRID APP DEVELOPMENT   **6**

Hybrid Web App, Benefits of Hybrid App, Criteria for creating Native App, Tools for creating Hybrid App, Cons of Hybrid App, Popular Hybrid App Development Frameworks, Ionic, Apache Cordova

UNIT IV CROSS-PLATFORM APP DEVELOPMENT USING REACT-NATIVE   **6**

What is Cross-platform App, Benefits of Cross-platform App, Criteria for creating Cross-platform App, Tools for creating Cross-platform App, Cons of   Cross-platform App, Popular Cross-platform App Development Frameworks, Flutter, Xamarin, React-Native, Basics of React Native, Native Components, JSX, State, Props

UNIT V NON-FUNCTIONAL CHARACTERISTICS OF APP FRAMEWORKS   **6**

Comparison of different Ap p frameworks, Build Performance, App Performance, Debugging capabilities, Time to Market, Maintainability, Ease of Development, UI/UX, Reusability

TOTAL NUMBER OF PERIODS INCLUDING LAB: 60

SAMPLE LIST OF EXPERIMENTS   **NUMBER OF PRACTICAL PERIODS: 30**

1. Using react native, build a cross platform application for a BMI calculator.
2. Build a cross platform application for a simple expense manager which allows entering expenses and income on each day and displays category wise weekly income and expense.
3. Develop a cross platform application to convert units from imperial system to metric system (km to miles, kg to pounds etc.,)
4. Design and develop a cross-platform application for day-to-day task (to-do) management.
5. Design an android application using Cordova for a user login screen with username, password, reset button and a submit button. Also, include header image and a label. Use layout managers.
6. Design and develop an android application using Apache Cordova to find and display the

107

current location of the user.

7. Write programs using Java to create Android application having Databases

For a simple library application.

For displaying books available, books lend, book reservation. Assume that student   information is available in a database which has been stored in a database server.

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Develop Native applications with GUI Components.

CO2   Develop hybrid applications with basic event handling.

CO3   Implement cross-platform applications with location and data storage capabilities.

TEXTBOOKS:

1.Head First Android Development, Dawn Griffiths, O'Reilly, 1st edition

2. Apache Cordova in Action, Raymond K. Camden, Manning. 2015

3.Full Stack React Native: Create beautiful mobile apps with JavaScript and React Native, Anthony Accomazzo, Houssein Djirdeh, Sophia Shoemaker, Devin Abbott, FullStack publishing

REFERENCES:

1.Android Programming for Beginners, John Horton, Packt Publishing, 2nd Edition

2. Native Mobile Development by Shaun Lewis, Mike Dunn

3. Building Cross-Platform Mobile and Web Apps for Engineers and Scientists: An Active Learning Approach, Pawan Lingras, Matt Triff, Rucha Lingras

4.Apache Cordova 4 Programming, John M Wargo, 2015

5.React Native Cookbook, Daniel Ward, Packt Publishing, 2nd Edition

Course Code	UI AND UX DESIGN	L	T	P	C
CSPE3XV13		**2**	**0**	**2**	**3**

COURSE OBJECTIVES:

1. To provide a sound knowledge in UI & UX
2. To understand the need for UI and UX
3. To understand the various Research Methods used in Design
4. To explore the various Tools used in UI & UX
5. Creating a wireframe and prototype

UNIT I FOUNDATIONS OF DESIGN 6

UI vs. UX Design - Core Stages of Design Thinking - Divergent and Convergent Thinking - Brainstorming and Game storming - Observational Empathy

UNIT II FOUNDATIONS OF UI DESIGN 6

Visual and UI Principles - UI Elements and Patterns - Interaction Behaviors and Principles – Branding - Style Guides

UNIT III FOUNDATIONS OF UX DESIGN 6

Introduction to User Experience - Why You Should Care about User Experience - Understanding User Experience - Defining the UX Design Process and its Methodology - Research in User Experience Design - Tools and Method used for Research - User Needs and its Goals - Know about Business Goals

UNIT IV WIREFRAMING, PROTOTYPING AND TESTING 6

Sketching Principles - Sketching Red Routes - Responsive Design – Wireframing - Creating Wireflows - Building a Prototype - Building High-Fidelity Mockups - Designing Efficiently with Tools - Interaction Patterns - Conducting Usability Tests - Other Evaluative User Research Methods - Synthesizing Test Findings - Prototype Iteration

UNIT V RESEARCH, DESIGNING, IDEATING, & INFORMATION ARCHITECTURE 6

Identifying and Writing Problem Statements - Identifying Appropriate Research Methods - Creating Personas - Solution Ideation - Creating User Stories - Creating Scenarios - Flow Diagrams - Flow Mapping - Information Architecture

TOTAL NUMBER OF PERIODS INCLUDING LAB: **60**

SAMPLE LIST OF EXPERIMENTS   **NUMBER OF PRACTICAL PERIODS:** **30**

1. Designing a Responsive layout for an societal application
2. Exploring various UI Interaction Patterns
3. Developing an interface with proper UI Style Guides
4. Developing Wire flow diagram for application using open-source software
5. Exploring various open-source collaborative interface Platform
6. Hands on Design Thinking Process for a new product
7. Brainstorming feature for proposed product

8.   Defining the Look and Feel of the new Project

9.   Create a Sample Pattern Library for that product (Mood board, Fonts, Colors based on UI principles)

10.   Identify a customer problem to solve

11.   Conduct end-to-end user research - User research, creating personas, Ideation process (User stories, Scenarios), Flow diagrams, Flow Mapping

12.   Sketch, design with popular tool and build a prototype and perform usability testing and identify improvements

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Build UI for user Applications
CO2   Evaluate UX design of any product or application
CO3   Demonstrate UX Skills in product development
CO4   Implement Sketching principles
CO5   Create Wireframe and Prototype

TEXTBOOKS:

1. Joel Marsh, "UX for Beginners", O'Reilly , 2022
2. Jon Yablonski, "Laws of UX using Psychology to Design Better Product & Services" O'Reilly 2021

REFERENCES:

1. Jenifer Tidwell, Charles Brewer, Aynne Valencia, "Designing Interface" 3rd Edition , O'Reilly 2020
2. Steve Schoger, Adam Wathan "Refactoring UI", 2018
3. Steve Krug, "Don't Make Me Think, Revisited: A Commonsense Approach to Web &Mobile", Third Edition, 2015
4. https://www.nngroup.com/articles/
5. https://www.interaction-design.org/literature.

Course Code	CLOUD SERVICES MANAGEMENT	L	T	P	C
CSPE3XV14		2	0	2	3

COURSE OBJECTIVES:

1. Introduce Cloud Service Management terminology, definition & concepts
2. Compare and contrast cloud service management with traditional IT service management.
3. Identify strategies to reduce risk and eliminate issues associated with the adoption of cloud services.
4. Select appropriate structures for designing, deploying, and running cloud-based services in a business environment.
5. Illustrate the benefits and drive the adoption of cloud-based services to solve real-world problems

UNIT I CLOUD SERVICE MANAGEMENT FUNDAMENTALS 6

Cloud Ecosystem, The Essential Characteristics, Basics of Information Technology Service Management and Cloud Service Management, Service Perspectives, Cloud Service Models, Cloud Service Deployment Models.

UNIT II CLOUD SERVICES STRATEGY 6

Cloud Strategy Fundamentals, Cloud Strategy Management Framework, Cloud Policy, Key Driver for Adoption, Risk Management, IT Capacity and Utilization, Demand and Capacity matching, Demand Queueing, Change Management, Cloud Service Architecture

UNIT III CLOUD SERVICE MANAGEMENT 6

Cloud Service Reference Model, Cloud Service Lifecycle, Basics of Cloud Service Design, Dealing with Legacy Systems and Services, Benchmarking of Cloud Services, Cloud Service Capacity Planning, Cloud Service Deployment and Migration, Cloud Marketplace, Cloud Service Operations Management.

UNIT IV CLOUD SERVICE ECONOMICS 6

Pricing models for Cloud Services, Freemium, Pay Per Reservation, Pay per User, Subscription-based Charging, Procurement of Cloud-based Services, Capex vs Opex Shift, Cloud service Charging, Cloud Cost Models.

UNIT V CLOUD SERVICE GOVERNANCE & VALUE 6

IT Governance Definition, Cloud Governance Definition, Cloud Governance Framework, Cloud Governance Structure, Cloud Governance Considerations, Cloud Service Model Risk Matrix, Understanding Value of Cloud Services, Measuring the Value of Cloud Services, Balanced Scorecard, Total Cost of Ownership.

TOTAL NUMBER OF PERIODS INCLUDING LAB: 60

SAMPLE LIST OF EXPERIMENTS   **NUMBER OF PRACTICAL PERIODS:** 30

1. Create a Cloud Organization in AWS/Google Cloud/or any equivalent Open-Source cloud softwares like OpenStack, Eucalyptus, Open Nebula with Role-based access control
2. Create a Cost-model for a web application using various services and do Cost-benefit analysis
3. Create alerts for usage of Cloud resources
4. Create Billing alerts for your Cloud Organization
5. Compare Cloud cost for a simple web application across AWS, Azure and GCP and suggest the best one.

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Exhibit cloud-design skills to build and automate business solutions using cloud technologies.
CO2   Possess Strong theoretical foundation leading to excellence and excitement towards adoption of cloud-based services.
CO3   Solve the real-world problems using Cloud services and technologies.

TEXTBOOKS:

1. Cloud Service Management and Governance: Smart Service Management in Cloud Era by Enamul Haque, Enel Publications
2. Cloud Computing: Concepts, Technology & Architecture by Thomas Erl, Ricardo Puttini, Zaigham Mohammad 2013
3. 3. Cloud Computing Design Patterns by Thomas Erl, Robert Cope, Amin Naserpour

REFERENCES:

1. Economics of Cloud Computing by Praveen Ayyappa, LAP Lambert Academic Publishing
2. Mastering Cloud Computing Foundations and Applications Programming Rajkumar Buyya, Christian Vechhiola, S. Thamarai Selvi

Course Code	DevOps	L	T	P	C
CSPE3XV15		2	0	2	3

COURSE OBJECTIVES:

1. To introduce DevOps terminology, definition & concepts
2. To understand the different Version control tools like Git, Mercurial
3. To understand the concepts of Continuous Integration/ Continuous Testing/ Continuous Deployment)
4. To understand Configuration management using Ansible
5. Illustrate the benefits and drive the adoption of cloud-based Devops tools to solve real world problems

UNIT I INTRODUCTION TO DEVOPS **6**

Devops Essentials - Introduction To AWS, GCP, Azure - Version control systems: Git and Github.

UNIT II COMPILE AND BUILD USING MAVEN & GRADLE **6**

Introduction, Installation of Maven, POM files, Maven Build lifecycle, Build phases (compile build,
test, package) Maven Profiles, Maven repositories (local, central, global), Maven plugins, Maven
create and build Artificats, Dependency management, Installation of Gradle, understand build using
Gradle

UNIT III CONTINUOUS INTEGRATION USING JENKINS **6**

Install & Configure Jenkins, Jenkins Architecture Overview, Creating a Jenkins Job, Configuring a Jenkins job, Introduction to Plugins, Adding Plugins to Jenkins, Commonly used plugins (Git Plugin, Parameter Plugin, HTML Publisher, Copy Artifact and Extended choice parameters). Configuring Jenkins to work with java, Git and Maven, Creating a Jenkins Build and Jenkins workspace.

UNIT IV CONFIGURATION MANAGEMENT USING ANSIBLE **6**

Ansible Introduction, Installation, Ansible master/slave configuration, YAML basics, Ansible modules, Ansible Inventory files, Ansible playbooks, Ansible Roles, adhoc commands in ansible

UNIT V BUILDING DEVOPS PIPELINES USING AZURE **6**

Create Github Account, Create Repository, Create Azure Organization, Create a new pipeline, Build a sample code, Modify azure-pipelines' file

TOTAL NUMBER OF PERIODS INCLUDING LAB: 60

SAMPLE LIST OF EXPERIMENTS　　　　　　　**NUMBER OF PRACTICAL PERIODS:**　30

1. Create Maven Build pipeline in Azure
2. Run regression tests using Maven Build pipeline in Azure
3. Install Jenkins in Cloud
4. Create CI pipeline using Jenkins
5. Create a CD pipeline in Jenkins and deploy in Cloud
6. Create an Ansible playbook for a simple web application infrastructure
7. Build a simple application using Gradle
8. Install Ansible and configure ansible roles and to write playbooks

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1　　Understand different actions performed through Version control tools like Git.

CO2　　Perform Continuous Integration and Continuous Testing and Continuous Deployment using Jenkins by building and automating test cases using Maven & Gradle.

CO3　　Ability to Perform Automated Continuous Deployment

CO4　　Ability to do configuration management using Ansible

CO5　　Understand to leverage Cloud-based DevOps tools using Azure DevOps

TEXTBOOKS:

1. Roberto Vormittag, "A Practical Guide to Git and GitHub for Windows Users: From Beginner to Expert in Easy Step-By-Step Exercises", Second Edition, Kindle Edition, 2016.

2. Jason Cannon, "Linux for Beginners: An Introduction to the Linux Operating System and Command Line", Kindle Edition, 2014

REFERENCES:

1. Hands-On Azure Devops: Cicd Implementation For Mobile, Hybrid, And Web Applications Using Azure Devops And Microsoft Azure: CICD Implementation for … DevOps and Microsoft Azure (English Edition) Paperback – 1 January 2020 by Mitesh Soni
2. Jeff Geerling, "Ansible for DevOps: Server and configuration management for humans", First Edition, 2015.
3. David Johnson, "Ansible for DevOps: Everything You Need to Know to Use Ansible for DevOps", Second Edition, 2016.
4. Mariot Tsitoara, "Ansible 6. Beginning Git and GitHub: A Comprehensive Guide to Version Control, Project Management, and Teamwork for the New Developer", Second Edition, 2019.
5. https://www.jenkins.io/user-handbook.pdf
6. https://maven.apache.org/guides/getting-started/

Course Code	WEB FRAMEWORKS	L	T	P	C
CSPE3XV16		2	0	2	3

COURSE OBJECTIVES:

1. To build scalable web applications using Angular
2. To import and export functionalities of modules using Angular
3. To create reusable UI components using React
4. To manage state of the application more efficiently using React Hook
5. To containerize the applications using Docker ad Kubernetes

UNIT I ANGULAR V 12   **6**

Introduction to Angular – Typescript (Arrays, Functions, classes) – JS vs TS – Angular CLI Installation – Components – Data Binding – Routing on Angular – Directives

UNIT II ANGULAR MODULES AND MATERIAL   **6**

Angular Modules – HTTP client, Forms Module – Angular Service Files – Dependancy Injection – Angular Material – Connecting Angular with Back End

UNIT III REACT V 18   **6**

Introduction to React – Setting development environment – create app – JSX syntax – properties and states – components – React routing – API request

        6

UNIT IV REACT HOOKS

React Hooks – useState – useEffect – useCallback – useMemo – useContext – useReducer – Introduction to React Native

        6

UNIT V CONTAINERIZATION

Introduction to Image and Container – Docker – Containers – Docker Images, Docker file, Docker Network – Docker Compose – Kubernetes

TOTAL NUMBER OF PERIODS INCLUDING LAB: 60

SAMPLE LIST OF EXPERIMENTS   **NUMBER OF PRACTICAL PERIODS:**   **30**

1. Project – Create an angular app with n components and add routing
2. Project – Add functionalities, validation and database with above components
3. Project – Create Login System using React
4. Project – Create Flight Management system

COURSE OUTCOMES:

On completion of this course, the students will be able to:
CO1   Build scalable web applications using Angular
CO2   Import and export functionalities of modules using Angular
CO3   Create reusable UI components using React

CO4   Manage state of the application more efficiently using React Hook
CO5   Containerize the applications using Docker ad Kubernetes
TEXTBOOKS:

1. Nate Murray, Felipe Coury, Ari Lerner, Carlos Taborda, " The Ng book — The Complete Book on Angular"
2. The Road to React, Robin Wieruch,2023.
3. The Docker Book: Containerization is the new virtualization, James Turnbull, 2014.
4. The Kubernetes Book, Nigel Poulton, 2023.

REFERENCES AND ONELINE RESOURCES:

1.   https://angular.io/docs

2.   https://react.dev/

3.   https://react.dev/reference/react

4.   https://docs.docker.com/

5.   https://kubernetes.io/docs/home/

Course Code	MIDDLE TIER TECHNOLOGIES	L	T	P	C
CSPE3XV17		2	0	2	3

COURSE OBJECTIVES:

1. To study the set of services that a middleware system constitutes of.
2. To understand how middleware facilitates the development of distributed applications in heterogeneous environments
3. To study how it helps to incorporate application portability, distributed application component interoperability and integration.
4. To learn the object-oriented middleware basics through the example of the following CORBA objects.
5. To understand the basics of Web services that is the most oft-used middleware technique

UNIT I CLIENT/ SERVER CONCEPTS   **6**

Client – Server – File Server, Database server, Group server, Object server, Web server. Middleware – General middleware – Service specific middleware. Client / Server Building blocks – RPC – Messaging – Peer – to- Peer.

UNIT II EJB ARCHITECTURE   **6**

EJB – EJB Architecture – Overview of EJB software architecture – View of EJB – Conversation – Building and Deploying EJBs – Roles in EJB.

UNIT III EJB APPLICATIONS   **6**

EJB Session Beans – EJB entity beans – EJB clients – EJB Deployment – Building an application with EJB.

UNIT IV CORBA   **6**

CORBA – Distributed Systems – Purpose – Exploring CORBA alternatives – Architecture overview – CORBA and networking model – CORBA object model – IDL – ORB – Building an application with CORBA.

UNIT V COM   **6**

COM – Data types – Interfaces – Proxy and Stub – Marshalling – Implementing Server / Client – Interface Pointers – Object Creation, Invocation , Destruction – Comparison COM and CORBA – Introduction to .NET – Overview of .NET architecture – Marshalling – Remoting.

TOTAL NUMBER OF PERIODS INCLUDING LAB: 60

SAMPLE LIST OF EXPERIMENTS   NUMBER OF PRACTICAL PERIODS: 30

1. Implementing a Simple File Server and Client Using Sockets.
2. Creating a Database Server Using JDBC and Client Communication.
3. Building and Deploying a Stateless Session Bean.
4. Developing and Deploying an EJB with JPA Integration.
5. Implementing EJB Session Beans and Client Interaction.
6. Building and Deploying EJB Entity Beans for Database Operations.
7. Creating a CORBA-Based Distributed Application.
8. Implementing and Testing CORBA IDL and ORB Communication.
9. Building a Simple COM Server and Client Application.
10. Implementing Interface Pointers and Object Creation in COM.

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Understanding the basic concepts of web application security and the need for it

CO2   Be acquainted with the process for secure development and deployment of web applications

CO3   Acquire the skill to design and develop Secure Web Applications that use Secure APIs

CO4   Be able to get the importance of carrying out vulnerability assessment and penetration testing

CO5   Acquire the skill to think like a hacker and to use hackers tool sets

TEXTBOOKS:

1. Robert Orfali, Dan Harkey and Jeri Edwards, "The Essential Client/Server Survival Guide", Galgotia Publications Pvt. Ltd., 2002. (Unit 1)
2. Tom Valesky,"Enterprise Java Beans",Pearson Education, 2002.(Unit 2 & 3)
3. Jason Pritchard,"COM and CORBA side by side", Addison Wesley,2000 (Unit 4 & 5)
4. Jesse Liberty, "Programming C#", 2nd Edition, O'Reilly Press, 2002. (Unit 5)

REFERENCES:

1. Mowbray, "Inside CORBA", Pearson Education, 2002.
2. Jeremy Rosenberger," Teach yourself CORBA in 14 days", Tec media, 2000

Course Code	WEB APPLICATION SECURITY	L	T	P	C
CSPE3XV18		2	0	2	3

COURSE OBJECTIVES:

1. To understand the fundamentals of web application security
2. To focus on wide aspects of secure development and deployment of web applications
3. To learn how to build secure APIs
4. To learn the basics of vulnerability assessment and penetration testing
5. To get an insight about Hacking techniques and Tools

UNIT I FUNDAMENTALS OF WEB APPLICATION SECURITY **6**

The history of Software Security-Recognizing Web Application Security Threats, Web Application Security, Authentication and Authorization, Secure Socket layer, Transport layer Security, Session Management-Input Validation

UNIT IISECURE DEVELOPMENT AND DEPLOYMENT **6**

Web Applications Security - Security Testing, Security Incident Response Planning, The Microsoft Security Development Lifecycle (SDL), OWASP Comprehensive Lightweight Application Security Process (CLASP), The Software Assurance Maturity Model (SAMM)

UNIT III SECURE API DEVELOPMENT **6**

API Security- Session Cookies, Token Based Authentication, Securing Natter APIs: Addressing threats with Security Controls, Rate Limiting for Availability, Encryption, Audit logging, Securing service-to-service APIs: API Keys , OAuth2, Securing Microservice APIs: Service Mesh, Locking Down Network Connections, Securing Incoming Requests.

UNIT IV VULNERABILITY ASSESSMENT AND PENETRATION TESTING **6**

Vulnerability Assessment Lifecycle, Vulnerability Assessment Tools: Cloud-based vulnerability scanners, Host-based vulnerability scanners, Network-based vulnerability scanners, Database-based vulnerability scanners, Types of Penetration Tests: External Testing, Web Application Testing, Internal Penetration Testing, SSID or Wireless Testing, Mobile Application Testing.

UNIT VHACKING TECHNIQUES AND TOOLS **6**

Social Engineering, Injection, Cross-Site Scripting(XSS), Broken Authentication and Session Management, Cross-Site Request Forgery, Security Misconfiguration, Insecure Cryptographic Storage, Failure to Restrict URL Access, Tools: Comodo, OpenVAS, Nexpose, Nikto, Burp Suite, etc.

TOTAL NUMBER OF PERIODS INCLUDING LAB: **60**

SAMPLE LIST OF EXPERIMENTS   **NUMBER OF PRACTICAL PERIODS: 30**

1. Install Wireshark and explore the various protocols
   a. Analyze the difference between HTTP vs HTTPS
   b. Analyze the various security mechanisms embedded with different protocols.
2. Identify the vulnerabilities using OWASP ZAP tool
3. Create simple REST API using python for following operation
   a. GET
   b. PUSH
   c. POST
   d. DELETE
4. Install Burp Suite to do following vulnerabilities:
   a. SQL injection
   b. cross-site scripting (XSS)
5. Attack the website using Social Engineering method

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Understanding the basic concepts of web application security and the need for it

CO2   Be acquainted with the process for secure development and deployment of web applications

CO3   Acquire the skill to design and develop Secure Web Applications that use Secure APIs

CO4   Be able to get the importance of carrying out vulnerability assessment and penetration testing

CO5   Acquire the skill to think like a hacker and to use hackers tool sets

TEXTBOOKS:

1. Andrew Hoffman, Web Application Security: Exploitation and Countermeasures for Modern Web Applications, First Edition, 2020, O'Reilly Media, Inc.

2. Bryan Sullivan, Vincent Liu, Web Application Security: A Beginners Guide, 2012, The McGraw-Hill Companies.

3. Neil Madden, API Security in Action, 2020, Manning Publications Co., NY, USA.

REFERENCES:

1. Michael Cross, Developer's Guide to Web Application Security, 2007, Syngress Publishing, Inc.

2. Ravi Das and Greg Johnson, Testing and Securing Web Applications, 2021, Taylor & Francis Group, LLC.

3. Prabath Siriwardena, Advanced API Security, 2020, Apress Media LLC, USA.

4. Malcom McDonald, Web Security for Developers, 2020, No Starch Press, Inc.

5. Allen Harper, Shon Harris, Jonathan Ness, Chris Eagle, Gideon Lenkey, and Terron Williams Grey Hat Hacking: The Ethical Hacker's Handbook, Third Edition, 2011, The McGraw-Hill Companies.

SOFTWARE TECHNOLOGIES

Course Code	SOFTWARE ENGINEERING	L	T	P	C
CSPE3XV21		3	0	0	3

COURSE OBJECTIVES:

1. To understand Software Engineering Lifecycle Models

2. To Perform software requirements analysis

3. To gain knowledge of the System Analysis and Design concepts using UML.

4. To understand software testing and maintenance approaches

5. To work on project management scheduling using DevOps

UNIT I SOFTWARE PROCESS AND AGILE DEVELOPMENT **6**

Introduction to Software Engineering, Software Process, Perspective and Specialized Process Models –Introduction to Agility-Agile process-Extreme programming-XP Process-Case Study.

UNIT II   REQUIREMENTS ANALYSIS AND SPECIFICATION **6**

Requirement analysis and specification – Requirements gathering and analysis – Functional and Non Functional requirements – Software Requirement Specification – Formal system specification – Finite State Machines – Petri nets – Object modelling using UML – Use case Model – Class diagrams – Interaction diagrams – Activity diagrams – Functional modelling – Data Flow Diagram- CASE TOOLS.

UNIT III   SOFTWARE DESIGN **6**

Software design – Design process – Design concepts – Coupling – Cohesion – Functional independence – Design patterns – Model-view-controller – Publish-subscribe – Adapter – Command
– Strategy – Observer – Proxy – Facade – Architectural styles – Layered - Client Server - Tiered - Pipe and filter- User interface design-Case Study.

UNIT IV SOFTWARE TESTING AND MAINTENANCE **6**

Testing – Unit testing – Black box testing– White box testing – Integration and System testing– Regression testing – Debugging - Program analysis – Symbolic execution – Model Checking-Case Study – Release Management.

UNIT V PROJECT MANAGEMENT **6**

Software Project Management- Software Configuration Management - Project Scheduling-DevOps: Motivation-Cloud as a platform-Operations- Deployment Pipeline: Overall

Architecture Building and Testing-Deployment- Tools- Case Study.

List of Practical Experiments:

1. Identify a software system that needs to be developed.
2. Document the Software Requirements Specification (SRS) for the identified system.
3. Identify use cases and develop the Use Case model.
4. Identify the conceptual classes and develop a Domain Model and also derive a Class Diagramfrom that.
5. Using the identified scenarios, find the interaction between objects and represent them using UML Sequence and Collaboration Diagrams
6. Draw relevant State Chart and Activity Diagrams for the same system.
7. Implement the system as per the detailed design
8. Test the software system for all the scenarios identified as per the use case diagram
9. Improve the reusability and maintainability of the software system by applying appropriate design patterns.
10. Implement the modified system and test it for various scenarios.

SUGGESTED DOMAINS FOR MINI-PROJECT:
1. Passport automation system.
2. Book bank
3. Exam registration
4. Stock maintenance system.
5. Online course reservation system
6. Airline/Railway reservation system
7. Software personnel management system
8. Credit card processing
9. e-book management system
10. Recruitment system
11. Foreign trading system
12. Conference management system
13. BPO management system
14. Library management system
15. Student information system

NUMBER OF PRACTICAL PERIODS: 30

TOTAL NUMBER OF PERIODS: 75

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Compare various Software Development Lifecycle Models
CO2   Evaluate project management approaches as well as cost and schedule
CO3   Perform formal analysis on specifications.
CO4   Use UML diagrams for analysis and design.
CO5   Architect and design using architectural styles and design patterns, and test the system

TEXTBOOKS:

1. Bernd Bruegge and Allen H. Dutoit, "Object-Oriented Software Engineering: Using UML, Patterns and Java", Third Edition, Pearson Education, 2009.

2. Roger S. Pressman, Object-Oriented Software Engineering: An Agile Unified Methodology, First Edition, Mc Graw-Hill International Edition, 2014.

REFERENCES:

1. Carlo Ghezzi, Mehdi Jazayeri, Dino Mandrioli, Fundamentals of Software Engineering,2nd edition, PHI Learning Pvt. Ltd., 2010.
2. Craig Larman, Applying UML and Patterns, 3rd ed, Pearson Education, 2005.
3. Len Bass, Ingo Weber and Liming Zhu, "DevOps: A Software Architect's Perspective", Pearson Education, 2016
4. Rajib Mall, Fundamentals of Software Engineering, 3rd edition, PHI Learning Pvt. Ltd., 2009.
5. Stephen Schach, Object-Oriented and Classical Software Engineering, 8th ed, McGraw-Hill

Course Code	SOFTWARE TESTING AND AUTOMATION	L	T	P	C
CSPE3XV22		2	0	2	3

COURSE OBJECTIVES:

1.   To understand the basics of software testing
2.   To learn how to do the testing and planning effectively
3.   To build test cases and execute them
4.   To focus on wide aspects of testing and understanding multiple facets of testing
5.   To get an insight about test automation and the tools used for test automation

UNIT I FOUNDATIONS OF SOFTWARE TESTING   6

Why do we test Software? Black-Box Testing and White-Box Testing, Software Testing Life Cycle, V-model of Software Testing, Program Correctness and Verification, Reliability versus Safety, Failures, Errors and Faults (Defects), Software Testing Principles, Program Inspections, Stages of Testing: Unit Testing, Integration Testing, System Testing

UNIT II TEST PLANNING   **6**

The Goal of Test Planning, High Level Expectations, Intergroup Responsibilities, Test Phases, Test Strategy, Resource Requirements, Tester Assignments, Test Schedule, Test Cases, Bug Reporting, Metrics and Statistics.

UNIT III TEST DESIGN AND EXECUTION   **6**

Test Objective Identification, Test Design Factors, Requirement identification, Testable Requirements, Modeling a Test Design Process, Modeling Test Results, Boundary Value Testing, Equivalence Class Testing, Path Testing, Data Flow Testing, Test Design Preparedness Metrics, Test Case Design Effectiveness, Model-Driven Test Design, Test Procedures, Test Case Organization and Tracking, Bug Reporting, Bug Life Cycle

UNIT IV ADVANCED TESTING CONCEPTS   **6**

Performance Testing: Load Testing, Stress Testing, Volume Testing, Fail-Over Testing, Recovery Testing, Configuration Testing, Compatibility Testing, Usability Testing, Testing the Documentation, Security testing, Testing in the Agile Environment, Testing Web and Mobile Applications.

UNIT V TEST AUTOMATION AND TOOLS   **6**

Automated Software Testing, Automate Testing of Web Applications, Selenium: Introducing Web Driver and Web Elements, Locating Web Elements, Actions on Web Elements, Different Web Drivers, Understanding Web Driver Events, Testing: Understanding Testing.xml, Adding Classes, Packages, Methods to Test, Test Reports.

TOTAL NUMBER OF PERIODS INCLUDING LAB: 60

SAMPLE LIST OF EXPERIMENTS   NUMBER OF PRACTICAL PERIODS: 30

1. Develop the test plan for testing an e-commerce web/mobile application (www.amazon.in).
2. Design the test cases for testing the e-commerce application
3. Test the e-commerce application and report the defects in it.
4. Develop the test plan and design the test cases for an inventory control system.
5. Execute the test cases against a client server or desktop application and identify the defects.
6. Test the performance of the e-commerce application.
7. Automate the testing of e-commerce applications using Selenium.
8. Integrate TestNG with the above test automation.
9. Mini Project:
    a. Build a data-driven framework using Selenium and TestNG
    b. Build Page object Model using Selenium and TestNG
    c. Build BDD framework with Selenium, TestNG and Cucumber

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Understand the basic concepts of software testing and the need for software testing

CO2   Understand the basic concepts of software testing and the need for software testing

CO3   Design effective test cases that can uncover critical defects in the application

CO4   Carry out advanced types of testing

CO5   Carry out advanced types of testing

TEXTBOOKS:

1. Yogesh Singh, "Software Testing", Cambridge University Press, 2012
2. Unmesh Gundecha, Satya Avasarala, "Selenium WebDriver 3 Practical Guide" - Second Edition 2018

REFERENCES:

1. Glenford J. Myers, Corey Sandler, Tom Badgett, The Art of Software Testing, 3rd Edition, 2012, John Wiley & Sons, Inc.
2. Ron Patton, Software testing, 2nd Edition, 2006, Sams Publishing
3. Paul C. Jorgensen, Software Testing: A Craftsman's Approach, Fourth Edition, 2014, Taylor & Francis Group.
4. Carl Cocchiaro, Selenium Framework Design in Data-Driven Testing, 2018, PacktPublishing.
5. Elfriede Dustin, Thom Garrett, Bernie Gaurf, Implementing Automated Software Testing, 2009, Pearson Education, Inc.
6. Satya Avasarala, Selenium WebDriver Practical Guide, 2014, Packt Publishing.
7. Varun Menon, TestNg Beginner's Guide, 2013, Packt Publishing.

Course Code	AGILE METHODOLOGIES	L	T	P	C
CSPE3XV23		2	0	2	3

COURSE OBJECTIVES:

1. To provide students with a theoretical as well as practical understanding of agile software development practices and how small teams can apply them to create high-quality software.

2. To provide a good understanding of software design and a set of software technologies and APIs.

3. To do a detailed examination and demonstration of Agile development and testing techniques.

4. To understand the benefits and pitfalls of working in an Agile team.

5. To understand Agile development and testing.

UNIT I AGILE METHODOLOGY **6**

Theories for Agile Management – Agile Software Development – Traditional Model vs. Agile Model - Classification of Agile Methods – Agile Manifesto and Principles – Agile Project Management – Agile Team Interactions – Ethics in Agile Teams - Agility in Design, Testing – Agile Documentations.

UNIT II AGILE PROCESSES **6**

Lean Production - SCRUM, Crystal, Feature Driven Development- Adaptive Software Development - Extreme Programming: Method Overview – Lifecycle – Work Products, Roles and Practices.

UNIT III AGILITY AND KNOWLEDGE MANAGEMENT **6**

Agile Information Systems – Agile Decision Making - Earl_S Schools of KM – Institutional Knowledge Evolution Cycle – Development, Acquisition, Refinement, Distribution, Deployment , Leveraging – KM in Software Engineering – Managing Software Knowledge – Challenges of Migrating to Agile Methodologies .

UNIT IV AGILITY AND REQUIREMENTS ENGINEERING **6**

Impact of Agile Processes in RE–Current Agile Practices – Variance – Overview of RE Using Agile – Managing Unstable Requirements – Requirements Elicitation – Agile Requirements Abstraction Model – Requirements Management in Agile Environment, Agile Requirements Prioritization – Agile Requirements Modeling and Generation – Concurrency in Agile Requirements Generation.

UNIT V AGILITY AND QUALITY ASSURANCE **6**

Agile Product Development – Agile Metrics – Feature Driven Development (FDD) – Financial and Production Metrics in FDD – Agile Approach to Quality Assurance - Test Driven Development – Agile Approach in Global Software Development.

TOTAL NUMBER OF PERIODS INCLUDING LAB: 60

SAMPLE LIST OF EXPERIMENTS   **NUMBER OF PRACTICAL PERIODS: 30**

1. Understand the background and driving forces for taking an Agile Approach to Software
2. development.
3. Build out a backlog and user stories.
4. To study automated build tool.
5. To study version control tool.
6. To study Continuous Integration tool.
7. Apply Design principle and Refactoring to achieve agility.
8. Perform Testing activities within an agile project.

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Realize the importance of interacting with business stakeholders in determining the requirements for a software system

CO2   Perform iterative software development processes: how to plan them, how to execute them

CO3   Point out the impact of social aspects on software development success.

CO4   Develop techniques and tools for improving team collaboration and software quality

CO5   Perform Software process improvement as an ongoing task for development teams.

TEXTBOOKS:

1. David J. Anderson and Eli Schragenheim, ─Agile Management for Software Engineering: Applying the Theory of Constraints for Business Results, Prentice Hall, 2003.

2. Hazza and Dubinsky, ─Agile Software Engineering, Series: Undergraduate Topics in Computer Science, Springer, 2009.

REFERENCES:

1. Craig Larman, ─Agile and Iterative Development: A Manager‗s Guide, Addison-Wesley, 2004.

2. Kevin C. Desouza, ─Agile Information Systems: Conceptualization, Construction, and Management, Butterworth-Heinemann, 2007.

Course Code	SOFTWARE USER INTERFACE DESIGN & ANALYSIS	L	T	P	C
CSPE3XV24		2	0	2	3

COURSE OBJECTIVES:

1. To understand the difference between UI and UX
2. To learn user-centered design principles.
3. To understand interaction design principles
4. To learn to create wireframes and prototypes
5. To explore advanced topics like responsive design.

Unit 1: INTRODUCTION TO UI/UX FUNDAMENTALS   6

Introduction to UI/UX concepts- Definition and differentiation of UI and UX- The Impact of UI/UX on Software Success- Understanding user expectations and behavior- User-Centric Design Principles- Importance of user-centric design- The Impact on User Satisfaction- Understanding user expectations and preferences.

Unit 2: DESIGN PRINCIPLES AND METHODOLOGIES   6

User-centered design principles- Definition and core principles of UCD- Historical context and evolution of UCD in design- Iterative Design Process- Usability heuristics- Definition and significance of usability heuristics -Application of Heuristics in Evaluation-

Unit 3: INTERACTION AND VISUAL DESIGN   6

Principles of interaction design- Definition and significance of interaction design principles- Usability and Learn ability- Fundamentals of visual design- Introduction to Visual Design- The psychology of visual elements and their impact on user perception- Color Theory and Application

Unit 4: PROTOTYPING AND USABILITY TESTING   6

Wire framing and prototyping- Definition and significance of wire framing and prototyping- Overview of their roles in the design process - Principles of Effective Wireframing- Usability testing methodologies

UNIT 5: ADVANCED TOPICS - RESPONSIVE DESIGN, ACCESSIBILITY, AND GUIDELINES   6

Responsive design principles- Definition and significance of responsive design - Principles of creating layouts with fluid grids - Techniques for making images responsive -Accessibility in UI design- UI guidelines and standards

TOTAL NUMBER OF PERIODS INCLUDING LAB: 60

SAMPLE LIST OF EXPERIMENTS   NUMBER OF PRACTICAL PERIODS: 30

1. Conduct heuristic evaluations

2. Apply user-centered design to a simple project.

3. Design interactive elements

4. Create visually appealing UI components

5. Conduct a usability test on a prototype

6. Implement responsive design in a project

7. Design an accessible user interface

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Design effective and user-centered interfaces.
CO2   Evaluate and improve user interfaces using usability heuristics.
CO3   Apply visual design principles for aesthetically pleasing interfaces.
CO4   Create prototypes and conduct usability tests.
CO5   Implement responsive design, ensure accessibility, and adhere to UI guidelines.

TEXTBOOKS:
1.
2.   "Don't Make Me Think" by Steve Krug
3.   "The Design of Everyday Things" by Don Norman
4.   "Seductive Interaction Design" by Stephen Anderson

REFERENCES:

1.   "Designing Interfaces" by Jenifer Tidwell
2.   "About Face: The Essentials of Interaction Design" by Alan Cooper
3.   "Universal Principles of Design" by William Lidwell, Kritina Holden, and Jill Butler

Course Code	SOFTWARE ARCHITECTURE AND DESIGN PATTERNS	L	T	P	C
CSPE3XV25		2	0	2	3

COURSE OBJECTIVES:

1. To define software architecture and its role in the development process
2. To understand the process of making architectural decisions.
3. To learn when and how to apply specific design patterns
4. To understand the principles and challenges of micro services architecture
5. To explore current trends and emerging paradigms in software architecture

UNIT 1: INTRODUCTION TO SOFTWARE ARCHITECTURE   **6**

Introduction to Software Architecture- Definition and significance of software architecture-Architectural Styles and Patterns - Overview of different architectural styles- Introduction to common design patterns.

UNIT 2: ARCHITECTURAL DECISION MAKING   **6**

Architectural Decision Process- Steps involved in making architectural decisions- Role of stakeholders in the decision making process- Trade-offs in Software Design- Evaluating trade- offs in terms of performance, scalability, and maintainability- Analyzing the impact of non-functional requirements on architecture.

UNIT 3: DESIGN PATTERNS IN DEPTH   **6**

Creational Design Patterns- Singleton, Factory Method, Abstract Factory patterns- When and how to use creational design patterns- **Structural and Behavioral Design Patterns**- Composite, Observer, Strategy patterns- Real-world examples illustrating the application of structural and behavioral design patterns.

Unit 4: ADVANCED TOPICS IN SOFTWARE ARCHITECTURE   **6**

Micro services Architecture-Principles of micro services architecture-Case studies on successful implementations-Event-Driven Architecture-Understanding events and event-driven systems-Implementing event-driven architectures in software systems.

Unit 5: EMERGING TRENDS AND FUTURE DIRECTIONS   **6**

Current Trends in Software Architecture-Overview of current trends, including serverless architecture, edge computing, etc-Case studies on how industry leaders are adapting to new trends-Continuous Learning and Professional Development-Importance of continuous learning in the field of software architecture-Resources, communities, and practices for staying updated and relevant.

                                TOTAL NUMBER OF PERIODS INCLUDING LAB: **60**

SAMPLE LIST OF EXPERIMENTS   **NUMBER OF PRACTICAL PERIODS:**   **30**

1. Analyze real-world applications and identify their architectural styles.

2. Discuss the impact of architectural decisions on system properties.

3. Analyze a complex scenario and make architectural decisions as a group

4. Present and defend the chosen architecture.

5. Conduct performance testing on a software system.

6. Hands-on coding session: Applying design patterns to a sample project

7. Identify areas for refactoring in existing code

8. Designing and implementing a micro services-based system

9. Testing and troubleshooting micro services interactions.

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Understand and Apply Software Architecture Concepts.
CO2   Understand the architectural decision-making process.
CO3   Apply design patterns to new projects and re-factor existing code.
CO4   Understand micro services architecture and its advantages and challenges.
CO5   Explore emerging trends in software architecture.

TEXTBOOKS:

1. "Design Patterns: Elements of Reusable Object-Oriented Software" by Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides
2. "Clean Architecture: A Craftsman's Guide to Software Structure and Design" by Robert C. Martin

REFERENCES:

1. 1."Microservices Patterns: With Examples in Java" by Chris Richardson
2. 2."Building Micro services" by Sam Newman
3. 3."Enterprise Integration Patterns: Designing, Building, and Deploying Messaging Solutions" by Gregor Hohpe and Bobby Woolf

Course Code	SOFTWARE QUALITY MANAGEMENT	L	T	P	C
CSPE3XV26		2	0	2	3

COURSE OBJECTIVES:

1. To define software quality and its significance in software development.
2. To define software testing and its role in the software development life cycle.
3. To learn software quality metrics and their significance
4. To understand software process improvement (SPI) and its goals..
5. To explore current trends and emerging practices in software quality management.

UNIT 1: INTRODUCTION TO SOFTWARE QUALITY MANAGEMENT   6

Overview of Software Quality- Definition and dimensions of software quality- Case studies on the consequences of poor software quality- Quality Management Principles- Evidence-Based Decision Making- Relationship Management- System Approach to Management- Factual Approach to Decision Making- Mutually Beneficial Supplier Relationships- Introduction to quality management frameworks (e.g., ISO 9001)

UNIT 2: SOFTWARE TESTING FUNDAMENTALS   6

Basics of Software Testing-Importance of software testing-Overview of testing levels and types-Test Planning and Strategy-Objective and Scope-Test Levels and Types-Test Deliverables-Resource Planning-Schedule and Timeline-Risk Assessment and Mitigation-Test Environment-Entry and Exit Criteria-Testing Tools and Techniques-Test Execution Strategy-Best Practices for Test Planning and Strategy.

UNIT 3: SOFTWARE QUALITY METRICS AND MEASUREMENT   6

Introduction to Quality Metrics-Defining key quality metrics-Relationship between metrics and project objectives-Measurement and Analysis-Principles of effective measurement-analysis in software quality-Informed decisions-Drive continuous improvement

UNIT 4: SOFTWARE PROCESS IMPROVEMENT   6

Introduction to Process Improvement-Basics of Software Process Improvement-Importance and benefits of SPI-Overview of common SPI models (e.g., CMMI, Six Sigma)- **Process Models and Frameworks-**Tailoring and Implementing Processes-Adapting process models-Best practices for implementing and sustaining process improvements.

UNIT 5: EMERGING TRENDS IN SOFTWARE QUALITY MANAGEMENT   6

Current Trends in Software Quality -Overview of current trends-test automation-AI in testing-importance of staying updated on industry trends-Continuous Learning and Professional Development-Importance of continuous learning-Resources, communities, and practices for staying updated and relevant.

TOTAL NUMBER OF PERIODS INCLUDING LAB: 60

SAMPLE LIST OF EXPERIMENTS   NUMBER OF PRACTICAL PERIODS: 30

1. Creating a basic Software Development Life Cycle (SDLC) model.

2. Building a Quality Assurance Plan for a sample project.

3. Writing and executing basic test cases for a simple application

4. Executing test cases manually on a sample application..

5. Introduction to test automation tools – setting up and running basic automated tests.

6. Simulating the defect life cycle using a bug tracking tool.

7. Root cause analysis and corrective action for identified defects.

8. Conducting performance tests on a web application

9. Exploring basic security testing techniques on a sample system.

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Define and Apply Software Quality Management Concepts
CO2   Understand and Apply Software Testing Fundamentals.
CO3   Apply Quality Metrics and Measurement.
CO4   Tailor and implement processes using different models and frameworks.
CO5   Develop a plan for continuous learning and professional development.

TEXTBOOKS:

1.   "Foundations of Software Testing" by Dorothy Graham and Erik Van Veenendaal

2.   "Software Engineering: A Practitioner's Approach" by Roger S. Pressman

REFERENCES:

1.   "Quality Software Management: Systems Thinking" by Gerald M. Weinberg

2.   "Improving Software Organizations: From Principles to Practice" by Louis Poulin

Course Code	SOFTWARE PROJECT MANAGEMENT	L	T	P	C
CSPE3XV27		2	0	2	3

COURSE OBJECTIVES:

1. To understand the Software Project Planning and Evaluation techniques.
2. To plan and manage projects at each stage of the software development life cycle (SDLC).
3. To learn about the activity planning and risk management principles.
4. To manage software projects and control software deliverables.
5. To develop skills to manage the various phases involved in project management and people management.

UNIT I PROJECT EVALUATION AND PROJECT PLANNING 6

Importance of Software Project Management – Activities - Methodologies – Categorization of Software Projects – Setting objectives – Management Principles – Management Control – Project portfolio Management – Cost-benefit evaluation technology – Risk evaluation .

UNIT II PROJECT LIFE CYCLE AND EFFORT ESTIMATION 6

Software process and Process Models – Choice of Process models - Rapid Application development – Agile methods – Dynamic System Development Method – Extreme Programming– Managing interactive processes – Basics of Software estimation – Effort and Cost estimation techniques – COSMIC Full function points - COCOMO II.

UNIT III ACTIVITY PLANNING AND RISK MANAGEMENT 6

Objectives of Activity planning – Project schedules – Activities – Sequencing and scheduling – Network Planning models – Formulating Network Model – Forward Pass & Backward Pass techniques – Critical path (CRM) method – Risk identification – Assessment – Risk Planning –Risk Management – – PERT technique – Monte Carlo simulation – Resource Allocation.

UNIT IV PROJECT MANAGEMENT AND CONTROL 6

Framework for Management and Control – Collection of data – Visualizing progress – Cost monitoring – Earned Value Analysis – Prioritizing Monitoring – Project tracking – Change Control – Software Configuration Management.

UNIT V STAFFING IN SOFTWARE PROJECTS 6

Managing people – Organizational behavior – Best methods of staff selection – Motivation – The Oldham – Hackman job characteristic model – Stress – Health and Safety – Ethical and Professional concerns – Working in teams – Decision making – Organizational structures – Dispersed and Virtual teams.

TOTAL NUMBER OF PERIODS INCLUDING LAB: 60

SAMPLE LIST OF EXPERIMENTS   NUMBER OF PRACTICAL PERIODS: 30

Software Architecture and Design Patterns Lab

Setting up project management tools (e.g., Jira, Trello) and introducing a sample project.
2. Creating a project plan with milestones, tasks, and resource allocation.
3. Conducting a project kickoff meeting and defining project scope.
4. Developing a Work Breakdown Structure (WBS) for a given project.
5. Identifying and analyzing project risks.
6. Developing a quality management plan and implementing quality control measures.
7. Assigning tasks and responsibilities using project management tools.
8. Using collaboration tools for effective communication and document sharing.
9. Implementing Agile methodologies for project management.

COURSE OUTCOMES:

On completion of this course, the students will be able to:
CO1   Understand Project Management principles while developing software.
CO2   Gain extensive knowledge about the basic project management concepts, framework and the process models.
CO3   Obtain adequate knowledge about software process models and software effort estimation techniques.
CO4   Estimate the risks involved in various project activities
CO5   Define the checkpoints, project reporting structure, project progress and tracking mechanisms using project management principles

TEXTBOOKS:

1. Bob Hughes, Mike Cotterell and Rajib Mall: Software Project Management – Fifth Edition, Tata McGraw Hill, New Delhi, 2012.

REFERENCES:

1. Robert K. Wysocki ―Effective Software Project Management – Wiley Publication, 2011. Walker Royce: ―Software Project Management- Addison-Wesley, 1998.
2. Gopalaswamy Ramesh, ―Managing Global Software Projects – McGraw Hill Education (India), Fourteenth Reprint 2013.

Course Code	HUMAN-COMPUTER INTERACTION	L	T	P	C
CSPE3XV28		2	0	2	3

COURSE OBJECTIVES

1. To learn the foundations of Human Computer Interaction.
2. To become familiar with the design technologies for individuals and persons with disabilities.
3. To be aware of mobile HCI.
4. To learn the guidelines for user interface.
5. Learn to develop meaningful user interface.

UNIT I FOUNDATIONS OF HCI   **6**

The Human: I/O channels – Memory – Reasoning and problem solving; The Computer: Devices – Memory – processing and networks; Interaction: Models – frameworks – Ergonomics – styles – elements – interactivity- Paradigms.

UNIT II DESIGN & SOFTWARE PROCESS   **6**

Interactive Design: Basics – process – scenarios – navigation – screen design – Iteration and prototyping. HCI in software process: Software life cycle – usability engineering – Prototyping in practice – design rationale. Design rules: principles, standards, guidelines, rules.

UNIT III MODELS AND THEORIES   **6**

HCI Models: Cognitive models: Socio-Organizational issues and stakeholder requirements – Communication and collaboration models-Hypertext, Multimedia and WWW.

UNIT IV MOBILE HCI   **6**

Mobile Ecosystem: Platforms, Application frameworks- Types of Mobile Applications: Widgets, Applications, Games- Mobile Information Architecture, Mobile 2.0, Mobile Design: Elements of Mobile Design, Tools.

UNIT V WEB INTERFACE DESIGN   **6**

Designing Web Interfaces – Drag & Drop, Direct Selection, Contextual Tools, Overlays, Inlays and Virtual Pages, Process Flow .

TOTAL NUMBER OF PERIODS INCLUDING LAB: 60

SAMPLE LIST OF EXPERIMENTS   NUMBER OF PRACTICAL PERIODS: 30

1. Setting up essential tools and resources for HCI design
2. Understanding the basics of user experience (UX) and user interface (UI) design.
3. Conducting user interviews and creating user personas.
4. Analyzing user behavior through usability testing and heuristic evaluation.
5. Creating paper prototypes for a given scenario.
6. Conducting usability tests on digital prototypes.
7. Analyzing usability testing results and making design refinements.

8. Applying interaction design principles to improve user interfaces.

9. Implementing responsive design for multiple devices.

10. Evaluating and improving accessibility in user interfaces.

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Design effective dialog for HCI
CO2   Design effective HCI for individuals and persons with disabilities
CO3   Assess the importance of user feedback
CO4   Explain the HCI implications for designing multimedia/ ecommerce/ e-learning Web sites.
CO5   Develop meaningful user interface.

TEXTBOOKS:

1. Alan Dix, Janet Finlay, Gregory Abowd, Russell Beale, ―Human Computer Interaction, 3rd Edition, Pearson Education, 2004

2. Brian Fling, ―Mobile Design and Development, First Edition, O'Reilly Media Inc., 2009

3. Bill Scott and Theresa Neil, ―Designing Web Interfaces, First Edition, O'Reilly, 2009.

Vertical III

CYBER SECURITY

Course Code	CRYPTOGRAPHY & INFORMATION SECURITY	L	T	P	C
CSPE3XV31		2	0	2	3

COURSE OBJECTIVES:

1. To provide deeper understanding into cryptography, its application to network security, threats/vulnerabilities to networks and countermeasures.

2. To explain various approaches to Encryption techniques, strengths of Traffic Confidentiality, Message Authentication Codes

3. To familiarize Digital Signature Standard and provide solutions for their issues.

4. To familiarize with cryptographic techniques for secure (confidential) communication of two parties over an insecure (public) channel; verification of the authenticity of the source of a message

5. To familiarize Authentication service , electronic mail security and web security.

UNIT - I INTRODUCTION

6

Security trends, The OSI Security Architecture, Security Attacks, Security Services and Security Mechanisms, A model for Network security. CLASSICAL ENCRYPTION TECHNIQUES: Symmetric Cipher Modes, Substitute Techniques, Transposition Techniques, Rotor Machines, Stenography.

UNIT - II BLOCK CIPHER AND DATA ENCRYPTION STANDARDS

6

Block Cipher Principles, Data Encryption Standards, the Strength of DES, Differential and Linear Crypt Analysis, Block Cipher Design Principles. ADVANCED ENCRYPTION STANDARDS: Evaluation Criteria for AES, the AES Cipher. MORE ON SYMMETRIC CIPHERS: Multiple Encryption, Triple DES, Block Cipher Modes of Operation, Stream Cipher and RC4. INTRODUCTION TO NUMBER THEORY: Prime Numbers, Fermat's and Euler's Theorem, Testing for Primality, The Chinese Remainder Theorem, Discrete logarithms

UNIT - III PUBLIC KEY CRYPTOGRAPHY AND RSA

6

Principles Public key crypto Systems, Diffie Hellman Key Exchange, the RSA algorithm, Key Management, Elliptic Curve Arithmetic, Elliptic Curve Cryptography. MESSAGE AUTHENTICATION AND HASH FUNCTIONS: Authentication Requirement, Authentication Function, Message Authentication Code, Hash Function, Security of Hash Function and MACs. HASH AND MAC ALGORITHM: Secure Hash Algorithm, Whirlpool, HMAC, CMAC. DIGITAL SIGNATURE: Digital Signature, Authentication Protocol, Digital Signature Standard.

UNIT - IV AUTHENTICATION APPLICATION

6

Kerberos, X.509 Authentication Service, Public Key Infrastructure. EMAIL SECURITY: Pretty Good Privacy (PGP) and S/MIME. IP SECURITY: Overview, IP Security Architecture, Authentication Header, Encapsulating Security Payload, Combining Security Associations and Key Management.

UNIT - V WEB SECURITY

6

Requirements, Secure Socket Layer (SSL) and Transport Layer Security (TLS), Secure Electronic Transaction (SET), Intruders, Viruses and related threats. FIREWALL: Firewall Design principles, Trusted Systems.

TOTAL NUMBER OF PERIODS INCLUDING LAB: 60

SAMPLE LIST OF EXPERIMENTS   **NUMBER OF PRACTICAL PERIODS:**   **30**

1. Symmetric Key Encryption
2. Asymmetric Key Cryptography
3. Secure Communication Protocols
4. Cryptographic Applications
5. Post-quantum Cryptography
6. Cryptographic Attacks and Countermeasures
7. Cryptography in IoT
8. Quantum-Safe Cryptography

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Identify basic security attacks and services

CO2   Use symmetric and asymmetric key algorithms for cryptography

CO3   Design a security solution for a given application

CO4   Analyze Key Management techniques and importance of number Theory with Message Authentication Codes and Hash Functions work

CO5   Understanding of Authentication functions and Authentication Service and Electronic Mail Security.

TEXTBOOKS:

1. William Stallings (2006), Cryptography and Network Security: Principles and Practice, 4th edition, Pearson Education, India.

2. 2. William Stallings (2000), Network Security Essentials (Applications and Standards), Pearson Education, India.

REFERENCES:

1. Charlie Kaufman (2002), Network Security: Private Communication in a Public World, 2nd edition, Prentice Hall of India, New Delhi.

2. Atul Kahate (2008), Cryptography and Network Security, 2nd edition, Tata Mc Grawhill, India.

3. 3. Robert Bragg, Mark Rhodes (2004), Network Security: The complete reference, Tata Mc Grawhill, India.

Course Code	SECURITY AND PRIVACY IN CLOUD	L	T	P	C
CSPE3XV32		2	0	2	3

COURSE OBJECTIVES:

1. To Introduce Cloud Computing terminology, definition & concepts
2. To understand the security design and architectural considerations for Cloud
3. To understand the Identity, Access control in Cloud
4. To follow best practices for Cloud security using various design patterns
5. To be able to monitor and audit cloud applications for security

UNIT I FUNDAMENTALS OF CLOUD SECURITY CONCEPTS   **6**

Overview of cloud security- Security Services - Confidentiality, Integrity, Authentication, Nonrepudiation, Access Control - Basic of cryptography - Conventional and public-key cryptography, hash functions, authentication, and digital signatures.

UNIT II SECURITY DESIGN AND ARCHITECTURE FOR CLOUD   **6**

Security design principles for Cloud Computing - Comprehensive data protection - End-to-end access control - Common attack vectors and threats - Network and Storage - Secure Isolation Strategies - Virtualization strategies - Inter-tenant network segmentation strategies - Data Protection strategies: Data retention, deletion and archiving procedures for tenant data, Encryption, Data Redaction, Tokenization, Obfuscation, PKI and Key

UNIT III ACCESS CONTROL AND IDENTITY MANAGEMENT   **6**

Access control requirements for Cloud infrastructure - User Identification - Authentication and Authorization - Roles-based Access Control - Multi-factor authentication - Single Sign-on, Identity Federation - Identity providers and service consumers - Storage and network access control options - OS Hardening and minimization - Verified and measured boot - Intruder Detection and prevention.

UNIT IV CLOUD SECURITY DESIGN PATTERNS   **6**

Introduction to Design Patterns, Cloud bursting, Geo-tagging, Secure Cloud Interfaces, Cloud Resource Access Control, Secure On-Premise Internet Access, Secure External Cloud

UNIT V MONITORING, AUDITING AND MANAGEMENT   **6**

Proactive activity monitoring - Incident Response, Monitoring for unauthorized access, malicious traffic, abuse of system privileges - Events and alerts - Auditing – Record generation, Reporting and Management, Tamper-proofing audit logs, Quality of Services, Secure Management, User management, Identity management, Security Information and Event Management

TOTAL NUMBER OF PERIODS INCLUDING LAB: **60**

SAMPLE LIST OF EXPERIMENTS   **NUMBER OF PRACTICAL**   **30 PERIODS:**

1. Simulate a cloud scenario using Cloud Sim and run a scheduling algorithm not present in Cloud Sim
2. simulate resource management using cloud sim
3. simulate log forensics using cloud sim
4. simulate a secure file sharing using a cloud sim
5. Implement data anonymization techniques over the simple dataset (masking, k-anonymization,

etc)

6. Implement any encryption algorithm to protect the images
7. Implement any image obfuscation mechanism
8. Implement a role-based access control mechanism in a specific scenario
9. implement an attribute-based access control mechanism based on a particular scenario
10. Develop a log monitoring system with incident management in the cloud

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Understand the cloud concepts and fundamentals.
CO2   Explain the security challenges in the cloud.
CO3   Define cloud policy and Identity and Access Management.
CO4   Understand various risks and audit and monitoring mechanisms in the cloud.
CO5   Define the various architectural and design considerations for security in the cloud.

TEXTBOOKS:

1. Raj Kumar Buyya , James Broberg, andrzejGoscinski, "Cloud Computing:", Wiley 2013
2. Dave shackleford, "Virtualization Security", SYBEX a wiley Brand 2013.
3. 3. Mather, Kumaraswamy and Latif, "Cloud Security and Privacy", OREILLY 2011

REFERENCES:

1. Mark C. Chu-Carroll ―Code in the Cloud‖,CRC Press, 2011
2. Mastering Cloud Computing Foundations and Applications Programming RajkumarBuyya, Christian Vechhiola, S. ThamaraiSelvi

Course Code	WEB APPLICATION SECURITY	L	T	P	C
CSPE3XV33		**2**	**0**	**2**	**3**

COURSE OBJECTIVES:

1. To understand the fundamentals of web application security
2. To focus on wide aspects of secure development and deployment of web applications
3. To learn how to build secure APIs
4. To learn the basics of vulnerability assessment and penetration testing
5. To get an insight about Hacking techniques and Tools

UNIT I FUNDAMENTALS OF WEB APPLICATION SECURITY   **6**

The history of Software Security-Recognizing Web Application Security Threats, Web Application Security, Authentication and Authorization, Secure Socket layer, Transport layer Security, Session Management-Input Validation

UNIT IISECURE DEVELOPMENT AND DEPLOYMENT   **6**

Web Applications Security - Security Testing, Security Incident Response Planning, The Microsoft Security Development Lifecycle (SDL), OWASP Comprehensive Lightweight Application Security Process (CLASP), The Software Assurance Maturity Model (SAMM)

UNIT III   SECURE API DEVELOPMENT   **6**

API Security- Session Cookies, Token Based Authentication, Securing Natter APIs: Addressing threats with Security Controls, Rate Limiting for Availability, Encryption, Audit logging, Securing service-to-service APIs: API Keys , OAuth2, Securing Microservice APIs: Service Mesh, Locking Down Network Connections, Securing Incoming Requests.

UNIT IV   VULNERABILITY ASSESSMENT AND PENETRATION TESTING   **6**

Vulnerability Assessment Lifecycle, Vulnerability Assessment Tools: Cloud-based vulnerability scanners, Host-based vulnerability scanners, Network-based vulnerability scanners, Database-based vulnerability scanners, Types of Penetration Tests: External Testing, Web Application Testing, Internal Penetration Testing, SSID or Wireless Testing, Mobile Application Testing.

UNIT VHACKING TECHNIQUES AND TOOLS   **6**

Social Engineering, Injection, Cross-Site Scripting(XSS), Broken Authentication and Session Management, Cross-Site Request Forgery, Security Misconfiguration, Insecure Cryptographic Storage, Failure to Restrict URL Access, Tools: Comodo, OpenVAS, Nexpose, Nikto, Burp Suite, etc.

TOTAL NUMBER OF PERIODS INCLUDING LAB: 60

SAMPLE LIST OF EXPERIMENTS   NUMBER OF PRACTICAL PERIODS: 30

1. Install wireshark and explore the various protocols
   a. Analyze the difference between HTTP vs HTTPS
   b. Analyze the various security mechanisms embedded with different protocols.
2. Identify the vulnerabilities using OWASP ZAP tool
3. Create simple REST API using python for following operation
   a. GET
   b. PUSH
   c. POST

d.   DELETE
4.   Install Burp Suite to do following vulnerabilities:
      b.SQL injection
      a. cross-site scripting (XSS)
5.   Attack the website using Social Engineering method

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Understanding the basic concepts of web application security and the need for it
CO2   Be acquainted with the process for secure development and deployment of web applications

CO3   Acquire the skill to design and develop Secure Web Applications that use Secure APIs
CO4   Be able to get the importance of carrying out vulnerability assessment and penetration testing
CO5   Acquire the skill to think like a hacker and to use hackers tool sets

TEXTBOOKS:

1.   Andrew Hoffman, Web Application Security: Exploitation and Countermeasures for Modern Web Applications, First Edition, 2020, O'Reilly Media, Inc.
2.   Bryan Sullivan, Vincent Liu, Web Application Security: A Beginners Guide, 2012, The McGraw-Hill Companies.
3.   Neil Madden, API Security in Action, 2020, Manning Publications Co., NY, USA.

REFERENCES:

1.   Michael Cross, Developer's Guide to Web Application Security, 2007, Syngress Publishing, Inc.

2.   Ravi Das and Greg Johnson, Testing and Securing Web Applications, 2021, Taylor & Francis Group, LLC.

3.   Prabath Siriwardena, Advanced API Security, 2020, Apress Media LLC, USA.

4.   Malcom McDonald, Web Security for Developers, 2020, No Starch Press, Inc.

5.   Allen Harper, Shon Harris, Jonathan Ness, Chris Eagle, Gideon Lenkey, and Terron Williams Grey Hat Hacking: The Ethical Hacker's Handbook, Third Edition, 2011, The McGraw-Hill Companies.

Course Code	SOCIAL NETWORK SECURITY	L	T	P	C
CSPE3XV34		2	0	2	3

COURSE OBJECTIVES:

1.   To develop semantic web related simple applications
2.   To explain Privacy and Security issues in Social Networking
3.   To explain the data extraction and mining of social networks
4.   To discuss the prediction of human behavior in social communities
5.   To describe the Access Control, Privacy and Security management of social networks

UNIT I FUNDAMENTALS OF SOCIAL NETWORKING **6**

Introduction to Semantic Web, Limitations of current Web, Development of Semantic Web, Emergence of the Social Web, Social Network analysis, Development of Social Network Analysis, Key concepts and measures in network analysis, Historical overview of privacy and security, Major paradigms, for understanding privacy and security, Common security threats in social networks and case studies of notable security incidents.

UNIT II SECURITY ISSUES IN SOCIAL NETWORKS **6**

Principles of secure user authentication, Multifactor authentication, Access control mechanisms and user permissions, SSL/TLS protocols for secure data transmission, End-to-end encryption in messaging apps, The evolution of privacy and security concerns with networked technologies, Contextual influences on privacy attitudes and behaviors, Anonymity in a networked world.

UNIT III EXTRACTION AND MINING IN SOCIAL NETWORKING DATA **6**

Extracting evolution of Web Community from a Series of Web Archive, Detecting communities in social networks, Definition of community, Evaluating communities, Methods for community detection and mining, Applications of community mining algorithms, Tools for detecting communities social network infrastructures and communities, Big data and Privacy, User education and awareness , regulatory compliance.

UNIT IV PREDICTING HUMAN BEHAVIOR AND PRIVACY ISSUES **6**

Understanding and predicting human behavior for social communities, User data Management, Inference and Distribution, Enabling new human experiences, Reality mining, Context, Awareness, Privacy in online social networks, Trust in online environment, What is Neo4j, Nodes, Relationships, Properties, Secure third-party integrations.

UNIT V ACCESS CONTROL, PRIVACY AND IDENTITY MANAGEMENT **6**

Understand the access control requirements for Social Network, Enforcing Access Control Strategies, Authentication and Authorization, Roles-based Access Control, Host, storage and network access control options, Firewalls, Authentication, and Authorization in Social Network, Identity & Access Management, Single Sign-on, Identity Federation, Identity providers and service consumers, The role of Identity provisioning. Emerging trends in social network security.

TOTAL NUMBER OF PERIODS INCLUDING LAB: **60**

SAMPLE LIST OF EXPERIMENTS   NUMBER OF PRACTICAL PERIODS: **30**

1. Design own social media application
2. Create a Network model using Neo4j
3. Read and write Data from Graph Database
4. Find "Friend of Friends" using Neo4j

5. Implement secure search in social media
6. Create a simple Security & Privacy detector

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Develop semantic web related simple applications
CO2   Address Privacy and Security issues in Social Networking
CO3   Explain the data extraction and mining of social networks
CO4   Discuss the prediction of human behavior in social communities
CO5   Describe the applications of social networks

TEXTBOOKS:

1. Peter Mika, "Social Networks and the Semantic Web, First Edition, Springer 2007.
2. BorkoFurht, "Handbook of Social Network Technologies and Application, First Edition, Springer, 2010.
3. Learning Neo4j 3.x "Second Edition By Jérôme Baton, Rik Van Bruggen, Packt publishing

REFERENCES:

1. Easley D. Kleinberg J., "Networks, Crowds, and Markets – Reasoning about a Highly Connected World", Cambridge University Press, 2010.
2. Jackson, Matthew O., "Social and Economic Networks", Princeton University Press, 2008.
3. GuandongXu ,Yanchun Zhang and Lin Li, "Web Mining and Social Networking – Techniques and applications", First Edition, Springer, 2011.

Course Code	DIGITAL FORENSICS AND MALWARE ANALYSIS	L	T	P	C
CSPE3XV35		2	0	2	3

COURSE OBJECTIVES:

1. Learn preventive measures to safeguard digital systems and information against cyber threats
2. Develop policies specific to digital crime and focusing on digital investigation
3. Gain a comprehensive understanding of the methodologies, tools, and techniques used in investigating various types of cybercrimes
4. Understand scope of the malware borne cyber-attacks, various malware types, and platform-specific variations of malware
5. Explain the basic signs of malware infection and signs of intrusion from a security analyst's point of view

UNIT I DIGITAL FORENSIC TECHNIQUE AND TRACES   **6**

File System Analysis - Network Packet Analysis - Memory Forensics - Digital Evidence Collection - Mobile Device Forensics Techniques - Cloud Forensics - Big Data Analytics in Forensics - Incident Response and Timeline Analysis.

UNIT II INTRODUCTION TO DIGITAL CRIME AND INVESTIGATION   **6**

Digital Crime Overview - Legal and Ethical Frameworks - Digital Forensic Tools and Techniques - Incident Response Fundamentals - Digital Evidence Collection and Preservation - Cybersecurity Basics - Digital Investigation Methodologies - Cybercrime Trends and Emerging Threats.

UNIT III INTERNET BASED INVESTIGATIONS   **6**

Social Media Investigations - Open-Source Intelligence (OSINT) - Email and Communication Tracing - Dark Web Investigations - Online Fraud Investigations - Cyber Threat Intelligence - Digital Copyright and Intellectual Property Investigations - Online Extremism and Radicalization Investigations.

UNIT IV INTRODUCTION TO MALWARE ANALYSIS   **6**

Malware classification- types- and platform specific issues with malware- Intrusion into IT and operational network (OT) and their signs – Basics of malware detection – Intro to linux- Networks.

UNIT V BASIC MALWARE ANALYSIS   **6**

Manual Malware Infection analysis - signature-based malware detection and classification – pros and cons- and need for machine learning based techniques- Advanced Techniques Malware Analysis.

TOTAL NUMBER OF PERIODS INCLUDING LAB: **60**

SAMPLE LIST OF EXPERIMENTS   **NUMBER OF PRACTICAL PERIODS:**   **30**

1. Live Case Studies Open Source Forensic Tools
2. Disk Forensics and Data Recovery
3. Steganography
4. Key loggers
5. Network monitors
6. Flowchart management
7. Upload Malware to Virustotal - Analyze the file Lab01-02.exe - [Refer Lab 1-2 in Textbook] - Lab01-02.exe
8. This lab uses the file Lab01-04.exe. Analyze the file Lab01-04.exe. [Refer Lab 1-4 in Textbook] - Lab01-04.exe
9. Analyze the malware found in the file Lab05-01.dll using only IDA Pro [Refer Lab 5-1 In Textbook] - Lab05-01.exe

10. Analyze the malware found in the file Lab07-02.exe [Refer Lab 07-02 in Textbook] - Lab07-02.exe

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Clarify the foundational principles of digital forensics, unravelling the core concepts essential to the field.

CO2   Evaluate and delineate strategies for managing risks associated with digital forensic investigations.

CO3   Elaborate on the basic tools and techniques utilized in the investigation of various types of cybercrimes.

CO4   Explain and introduce malware analysis efficiently.

CO5   Explain basic malware analysis and demonstrate some basic techniques.

TEXTBOOKS:

1. Andre Arnes, "Digital Forensics", Wiley, 2018.
2. Chuck Easttom, "An In-depth Guide to Mobile Device Forensics", First Edition, CRC Press, 2022.
3. Practical Malware Analysis By, Michael Siroski and Andrew Honig - Link

REFERENCES:

1. Vacca, J, *Computer Forensics, Computer Crime Scene Investigation*, 2nd Ed, Charles River Media, 2005, ISBN: 1-58450-389.
2. Digital Evidence and Computer Crime: Forensic Science, Computers, and the Internet, 3rd Edition Author: Eoghan Casey Publisher: Academic Press ISBN: 9780123742681.
3. Dynamic Malware Analysis in the Modern Era—A State of the Art Survey - Link

Online Reference:

4. https://www.coursera.org/specializations/computerforensics
5. https://www.youtube.com/watch?v=u2zgEFm5RHQ
6. Practical Malware analysis - Youtube

Course Code	ETHICAL HACKING	L	T	P	C
CSPE3XV36		2	0	2	3

COURSE OBJECTIVES:

1.   To understand the basics of computer based vulnerabilities.
2.   To explore different foot printing, reconnaissance and scanning methods.
3.   To expose the enumeration and vulnerability analysis methods.
4.   To understand hacking options available in Web and wireless applications.
5.   To explore the options for network protection and perform ethical hacking to expose the vulnerabilities.

UNIT I INTRODUCTION   **6**

Ethical Hacking Overview - Role of Security and Penetration Testers .- Penetration-Testing Methodologies- Laws of the Land - Overview of TCP/IP- The Application Layer - The Transport Layer - The Internet Layer - IP Addressing .- Network and Computer Attacks - Malware - Protecting Against Malware Attacks.- Intruder Attacks - Addressing Physical Security

UNIT II FOOT PRINTING, RECONNAISSANCE AND SCANNING NETWORKS   **6**

Footprinting Concepts - Footprinting through Search Engines, Web Services, Social Networking Sites, Website, Email - Competitive Intelligence - Footprinting through Social Engineering - Footprinting Tools - Network Scanning Concepts - Port-Scanning Tools - Scanning Techniques - Scanning Beyond IDS and Firewall

UNIT III ENUMERATION AND VULNERABILITY ANALYSIS   **6**

Enumeration Concepts - NetBIOS Enumeration – SNMP, LDAP, NTP, SMTP and DNS Enumeration - Vulnerability Assessment Concepts - Desktop and Server OS Vulnerabilities - Windows OS Vulnerabilities - Tools for Identifying Vulnerabilities in Windows- Linux OS Vulnerabilities-Vulnerabilities of Embedded Oss

UNIT IV SYSTEM HACKING   **6**

Hacking Web Servers - Web Application Components- Vulnerabilities - Tools for Web Attackers and Security Testers Hacking Wireless Networks - Components of a Wireless Network – Wardriving-Wireless Hacking - Tools of the Trade .

UNIT V NETWORK PROTECTION SYSTEMS   **6**

Access Control Lists. - Cisco Adaptive Security Appliance Firewall - Configuration and Risk Analysis Tools for Firewalls and Routers - Intrusion Detection and Prevention Systems - NetworkBased and Host-Based IDSs and IPSs - Web Filtering - Security Incident Response Teams – Honeypots.

TOTAL NUMBER OF PERIODS INCLUDING LAB: **60**

SAMPLE LIST OF EXPERIMENTS   NUMBER OF PRACTICAL PERIODS: **30**

1. Install Kali or Backtrack Linux / Metasploitable/ Windows XP

2. Practice the basics of reconnaissance.

3. Using FOCA / SearchDiggity tools, extract metadata and expanding the target list.

4. Aggregates information from public databases using online free tools like Paterva's Maltego.

5. Information gathering using tools like Robtex.

6. Scan the target using tools like Nessus.

7. View and capture network traffic using Wireshark.

8. Automate dig for vulnerabilities and match exploits using Armitage

FOCA : http://www.informatica64.com/foca.aspx. 134

Nessus : http://www.tenable.com/products/nessus.

Wireshark : http://www.wireshark.org.

Armitage : http://www.fastandeasyhacking.com/.

Kali or Backtrack Linux, Metasploitable, Windows XP

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   To express knowledge on basics of computer-based vulnerabilities

CO2   To gain understanding on different foot printing, reconnaissance and scanning methods.

CO3   To demonstrate the enumeration and vulnerability analysis methods

CO4   To gain knowledge on hacking options available in Web and wireless applications

CO5   To acquire knowledge on the options for network protection and perform ethical hacking to expose the vulnerabilities

TEXTBOOKS:

1. Michael T. Simpson, Kent Backman, and James E. Corley, Hands-On Ethical Hacking and Network Defense, Course Technology, Delmar Cengage Learning, 2010.

2. The Basics of Hacking and Penetration Testing - Patrick Engebretson, SYNGRESS, Elsevier, 2013.

3. 3. The Web Application Hacker's Handbook: Finding and Exploiting Security Flaws, Dafydd Stuttard and Marcus Pinto, 2011.

REFERENCES:

1. Black Hat Python: Python Programming for Hackers and Pentesters, Justin Seitz , 2014

Course Code	CRYPTOCURRENCY & BLOCKCHAIN TECHNOLOGIES	L	T	P	C
CSPE3XV37		2	0	2	3

COURSE OBJECTIVES:

1. To understand the basics of Blockchain
2. To learn Different protocols and consensus algorithms in Blockchain
3. To learn the Blockchain implementation frameworks
4. To experiment the Hyperledger Fabric, Ethereum networks
5. To understand the Blockchain Applications

UNIT I INTRODUCTION TO BLOCKCHAIN   **6**

Blockchain- Public Ledgers, Blockchain as Public Ledgers - Block in a Blockchain, TransactionsThe Chain and the Longest Chain - Permissioned Model of Blockchain, Cryptographic -Hash Function, Properties of a hash function-Hash pointer and Merkle tree.

UNIT II BITCOIN AND CRYPTOCURRENCY   **6**

A basic crypto currency, Creation of coins, Payments and double spending, FORTH – the precursor for Bitcoin scripting, Bitcoin Scripts , Bitcoin P2P Network, Transaction in Bitcoin Network, Block Mining, Block propagation and block relay

UNIT III BITCOIN CONSENSUS   **6**

Bitcoin Consensus, Proof of Work (PoW)- Hashcash PoW , Bitcoin PoW, Attacks on PoW ,monopoly problem- Proof of Stake- Proof of Burn - Proof of Elapsed Time - Bitcoin Miner, Mining Difficulty, Mining Pool-Permissioned model and use cases.

UNIT IV HYPERLEDGER FABRIC & ETHEREUM   **6**

Architecture of Hyperledger fabric v1.1- chain code- Ethereum: Ethereum network, EVM, Transaction fee, Mist Browser, Ether, Gas, Solidity. UNIT V BLOCKCHAIN APPLICATIONS 6 Smart contracts, Truffle Design and issue- DApps- NFT. Blockchain Applications in Supply Chain Management, Logistics, Smart Cities, Finance and Banking, Insurance - Case Study.

UNIT V BLOCKCHAIN APPLICATIONS   **6**

Smart contracts, Truffle Design and issue- DApps- NFT. Blockchain Applications in Supply Chain Management, Logistics, Smart Cities, Finance and Banking, Insurance,etc- Case Study.

TOTAL NUMBER OF PERIODS INCLUDING LAB: **60**

SAMPLE LIST OF EXPERIMENTS   **NUMBER OF PRACTICAL PERIODS:**   **30**

COURSE OUTCOMES:
On completion of this course, the students will be able to:
CO1   Understand emerging abstract models for Blockchain Technology
CO2   Identify major research challenges and technical gaps existing between theory and practice in the crypto currency domain.
CO3   It provides conceptual understanding of the function of Blockchain as a method of securing distributed ledgers.
CO4   Apply hyperledger Fabric and Ethereum platform to implement the Block chain Application.
CO5   To design and develop the Block chain Application.

TEXTBOOKS:

1. Bashir and Imran, Mastering Blockchain: Deeper insights into decentralization, cryptography, Bitcoin, and popular Blockchain frameworks, 2017.

2. Andreas Antonopoulos, "Mastering Bitcoin: Unlocking Digital Cryptocurrencies", O'Reilly, 2014.

3. Handbook of Research on Blockchain Technology, published by Elsevier Inc. ISBN: 9780128198162, 2020.

REFERENCES:

1. Daniel Drescher, "Blockchain Basics", First Edition, Apress, 2017.

2. Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller, and Steven Goldfeder. Bitcoin and cryptocurrency technologies: a comprehensive introduction. Princeton University Press, 2016.

3. Melanie Swan, "Blockchain: Blueprint for a New Economy", O'Reilly, 2015 4. Ritesh Modi, "Solidity Programming Essentials: A Beginner's Guide to Build Smart Contracts for Ethereum and Blockchain", Packt Publishing

Course Code	SECURITY AUDITING & COUNTER HACKING	L	T	P	C
CSPE3XV38	TECHNIQUES	2	0	2	3

COURSE OBJECTIVES:

1. Understand fundamentals, tools, and methodologies for security audits
2. Learn techniques to identify and mitigate network vulnerabilities
3. Explore methods to assess and secure web applications against hacking
4. Develop skills for detecting and responding to security incidents, including digital forensics
5. Acquire knowledge on advanced counter-hacking techniques and strategies for proactive security

UNIT I FUNDAMENTALS OF SECURITY AUDITING   **6**

Introduction to Cybersecurity, Principles of Security Auditing, Types of Security Audits, Security Auditing Tools, Methodologies for Security Assessments, Regulatory Compliance Standards, Security Policies and Procedures, Incident Response Planning, Security Documentation and Reporting

UNIT II NETWORK VULNERABILITY ASSESSMENT   **6**

Network Scanning Techniques, Identification and Classification of Vulnerabilities, Exploitation and Penetration Testing, Risk Assessment Methodologies, Remediation Strategies and Best Practices, Network Security Architecture, Firewalls and Intrusion Detection/Prevention Systems, Wireless Network Security, Security Assessments for IoT Devices

UNIT III WEB APPLICATION SECURITY TESTING   **6**

Overview of Web Application Architecture. Common Web Vulnerabilities (e.g., SQL Injection, XSS), Testing Methodologies (e.g., OWASP Top 10), Secure Coding Practices, Web Application Firewall (WAF) Usage, Session Management and Authentication, Content Security Policy (CSP)
API Security Considerations, Mobile Application Security Testing

UNIT IV INCIDENT RESPONSE AND DIGITAL FORENSICS   **6**

Incident Detection and Classification, Incident Response Strategies and Frameworks, Basics and Principles of Digital Forensics, Forensic Tools and Analysis Techniques, Chain of Custody and Legal Considerations, Memory Forensics and Disk Analysis, Network Forensics, Malware Analysis and Reverse Engineering, Incident Documentation and Reporting

UNIT V ADVANCED COUNTER-HACKING STRATEGIES   **6**

Proactive Security Measures and Threat Hunting, Threat Intelligence and Analysis, Offensive Countermeasures and Red Teaming, Security Awareness and Training Programs, Security Automation and Orchestration, Cloud Security Considerations, Blockchain Security, Emerging Trends in Cybersecurity Defense, Ethical and Legal Aspects of Counter-Hacking.

TOTAL NUMBER OF PERIODS INCLUDING LAB: 60

SAMPLE LIST OF EXPERIMENTS   NUMBER OF PRACTICAL PERIODS: 30

1. Analyze and critique a security audit report, identifying vulnerabilities and suggesting mitigation strategies.
2. Execute a simulated network vulnerability assessment, documenting findings and proposing remediation measures.

3. Conduct a hands-on web application penetration test, addressing identified vulnerabilities and suggesting security enhancements.
4. Develop an incident response plan for a hypothetical security incident, outlining detection, response, and recovery steps.
5. Analyze a digital forensics case study, presenting findings and recommendations for legal proceedings.
6. Research and analyze threat intelligence data, creating a comprehensive report on potential risks and proactive security measures.
7. Participate in a red teaming exercise to simulate advanced cyber threats, gaining insights into offensive counter-hacking strategies.
8. Design a comprehensive security awareness training program, incorporating best practices and tailored content for end-users.
9. Assess the security of a cloud infrastructure, identifying vulnerabilities and proposing strategies for secure cloud adoption.
10. Research and write a whitepaper on blockchain security, addressing potential threats and proposing protective measures.

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1 Grasp cybersecurity principles and apply security auditing methodologies effectively.

CO2 Master network scanning, vulnerability identification, and risk assessment techniques.

CO3 Understand web application vulnerabilities, testing methodologies, and secure coding practices

CO4 Execute incident response strategies and perform digital forensics with precision

CO5 Implement proactive security measures, threat intelligence analysis, and offensive countermeasures

TEXTBOOKS:

1. "The Web Application Hacker's Handbook" by Dafydd Stuttard and Marcus Pinto (2020, 2nd)
2. "Security Engineering" by Ross J. Anderson (2020, 3rd)
3. "Cybersecurity – Attack and Defense Strategies" by Yuri Diogenes and Erdal Ozkaya (2020, 1st)
4. "Blockchain Basics" by Daniel Drescher (2017, 1st)
5. "Network Security Essentials" by William Stallings (2017, 6th)
6. "Incident Response & Computer Forensics" by Jason T. Luttgens, Matthew Pepe, and Kevin M (2014, 3rd)
7. "Threat Modeling: Designing for Security" by Adam Shostack (2014, 1st)

REFERENCES:

1. Dafydd Stuttard, Marcus Pinto. "The Web Application Hacker's Handbook" (2020, 2nd)
2. Ross J. Anderson. "Security Engineering" (2020, 3rd)
3. Yuri Diogenes, Erdal Ozkaya. "Cybersecurity – Attack and Defense Strategies" (2020, 1st)
4. Daniel Drescher. "Blockchain Basics" (2017, 1st)
5. William Stallings. "Network Security Essentials" (2017, 6th)

ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

Course Code	DATA ANALYTICS AND VISUALIZATION	L	T	P	C
CSPE3XV41		2	0	2	3

COURSE OBJECTIVES:

1. To understand the data science fundamentals and process.
2. To learn to describe the relationship between data.
3. To utilize present and interpret data using Python libraries for Data Wrangling and data visualization.
4. To study the basic inferential statistics, sampling distribution and processes in data analytics
5. To understand the data analytics techniques and apply descriptive data analytics techniques

UNIT I INTRODUCTION   **6**

Data Science: Benefits and uses – facets of data - Data Science Process: Overview – Defining research goals – Retrieving data – Data preparation - Exploratory Data analysis – build the model– presenting findings and building applications - Data Mining - Data Warehousing – Basic Statistical descriptions of Data.

UNIT II DESCRIBING DATA   **6**

Basics of Numpy arrays –aggregations –computations on arrays –comparisons, masks, boolean logic – fancy indexing – structured arrays – Data manipulation with Pandas – data in Types of Data - Types of Variables -Describing Data with Tables and Graphs –Describing Data with Averages - Describing Variability - Normal Distributions and Standard (z) Scores.

DESCRIBING RELATIONSHIPS: Correlation –Scatter plots –correlation coefficient for quantitative data –computational formula for correlation coefficient – Regression –regression line –least squares regression line – Standard error of estimate – interpretation of r2 –multiple regression equations –regression towards the mean

UNIT III PYTHON LIBRARIES FOR DATA WRANGLING   **6**

Indexing and selection – operating on data – missing data – Hierarchical indexing – combining datasets – aggregation and grouping – pivot tables .

DATA VISUALIZATION: Importing Matplotlib – Line plots – Scatter plots – visualizing errors – density and contour plots – Histograms – legends – colors – subplots – text and annotation – customization – three-dimensional plotting - Geographic Data with Basemap - Visualization with Seaborn.

UNIT IV DESCRIPTIVE ANALYTICS AND INFERENTIAL STATISTICS   **6**

DESCRIPTIVE ANALYTICS - Frequency distributions – Outliers –interpreting distributions – graphs – averages - describing variability – interquartile range – variability for qualitative and ranked data - Normal distributions – z scores –correlation – scatter plots – regression – regression line – least squares regression line – standard error of estimate – interpretation of r2 – multiple regression equations – regression toward the mean.

INFERENTIAL STATISTICS - populations – samples – random sampling – Sampling distribution- standard error of the mean - Hypothesis testing – z-test – z-test procedure – decision rule – calculations – decisions – interpretations - one-tailed and two-tailed tests –

Estimation – point estimate – confidence interval – level of confidence – effect of sample size.

UNIT V ANALYSIS OF VARIANCE AND PREDICTIVE ANALYTICS   **6**

ANALYSIS OF VARIANCE - T-test for one sample – sampling distribution of t – t-test procedure – t-test for two independent samples – p-value – statistical significance – t-test for two related samples. F-test – ANOVA – Two-factor experiments – three f-tests – two-factor ANOVA – Introduction to chi-square tests.

PREDICTIVE ANALYTICS - Linear least squares – implementation – goodness of fit – testing a linear model – weighted resampling. Regression using Stats Models – multiple regression – nonlinear relationships – logistic regression – estimating parameters – Time series analysis – moving averages – missing values – serial correlation – autocorrelation. Introduction to survival analysis.

TOTAL NUMBER OF PERIODS INCLUDING LAB: 30

SAMPLE LIST OF EXPERIMENTS   **NUMBER OF PRACTICAL**   **30**
PERIODS:
1. Download, install and explore the features of NumPy, SciPy, Jupiter, Stats models and Pandas packages, Scipy, Matplotlib, Pandas, stat models, seaborn, plotly, bokeh
2. Working with NumPy arrays
3. Working with Pandas data frames
4. Reading data from text files, Excel and the web and exploring various commands for doing descriptive analytics on the Iris data set.
5. Use the diabetes data set from UCI and Pima Indians Diabetes data set for performing the following:
6. Univariate analysis: Frequency, Mean, Median, Mode, Variance, Standard Deviation, Skewness and Kurtosis.
7. Bivariate analysis: Linear and logistic regression modeling
8. Multiple Regression analysis
9. Also compare the results of the above analysis for the two data sets.
10. Apply and explore various plotting functions on UCI data sets.
11. Normal curves
12. Perform Z-test
13. Perform T-test
14. Perform ANOVA
15. Building and validating linear models
16. Building and validating logistic models
17. Time series analysis

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Define the data science process Understand different types of data descriptions for data science process

CO2   Gain knowledge on relationships between data and use the Python Libraries for Data Wrangling

CO3	Apply visualization Libraries in Python to interpret and explore data
CO4	Perform various statistical analyses to make statistical inferences and explain the end-to-end data analytics pipeline
CO5	Build, validate and communicate data analytical models for complex engineering problems

TEXTBOOKS:

1. David Cielen, Arno D. B. Meysman, and Mohamed Ali, "Introducing Data Science", Manning Publications, 2016.
2. Robert S. Witte and John S. Witte, "Statistics", Eleventh Edition, Wiley Publications, 2017. Padeepz App Padeepz App 69
3. Jake VanderPlas, "Python Data Science Handbook", O'Reilly, 2016.

REFERENCES:

1. Allen B. Downey, "Think Stats: Exploratory Data Analysis in Python", Green Tea Press,2014.
2. Peter Bruce, Andrew Bruce, and Peter Gedek, "Practical Statistics for Data Scientists", Second Edition, O'Reilly Publishers, 2020.
3. Charles R. Severance, "Python for Everybody: Exploring Data in Python 3", Shroff Publishers, 2017
4. Bradley Efron and Trevor Hastie, "Computer Age Statistical Inference", Cambridge University Press, 2016.

Course Code	**MATHEMATICS OF MACHINE LEARNING**	L	T	P	C
CSPE3XV42		2	0	2	3

COURSE OBJECTIVES:

1. Understand fundamental linear algebra concepts, including vector spaces, matrices, eigenvalues, and eigenvectors.

2. Gain proficiency in advanced calculus techniques, including gradient-based optimization, to train and fine-tune learning models effectively for optimal performance.

3. Explore the role of probability and statistics in learning and understanding their significance in model training, uncertainty estimation, and probabilistic modeling.

4. Apply mathematical models through hands-on projects, implementing machine learning models.

5. Give exposure to the deep learning models and analyse their performance using mathematical tools.

UNIT I LINEAR ALGEBRA, MATRIX, AND ANALYTICAL GEOMETRY   6

Introduction and Motivation - Linear Algebra, Systems of Linear Equations, Matrices, Solving Systems of Linear Equations, Vector Spaces, Linear Independence, Basis and Rank, Linear Mappings, Affine Spaces, Analytic Geometry, Norms, Inner Products, Lengths and Distances, Angles and Orthogonality, Orthonormal Basis, Orthogonal Complement, Inner Product of Functions, Orthogonal Projections, Rotations.

UNIT II MATRIX DECOMPOSITION AND VECTOR CALCULUS   6

Matrix Decompositions, Determinant and Trace, Eigenvalues and Eigenvectors, Cholesky Decomposition, Eigen-decomposition and Diagonalization, Singular Value Decomposition, Matrix Approximation, Vector Calculus, Differentiation of Univariate, Partial Differentiation and Gradients, Gradients of Vector-Valued Functions, Gradients of Matrices, Useful Identities for Computing Gradients, Backpropagation and Automatic Differentiation, Higher-Order Derivatives, Linearization, and Multivariate Taylor Series.

UNIT III PROBABILITY DISTRIBUTIONS AND RISK MINIMISATION   6

Probability and Distributions - Construction of a Probability Space-Discrete and Continuous Probabilities, Sum Rule, Product Rule, and Bayes' Theorem, Independence and Gaussian Distribution - Conjugacy and the Exponential Family, Change of Variables/Inverse Transform, Continuous Optimization, Optimization Using Gradient Descent, Constrained Optimization and Lagrange Multipliers, Convex Optimization. Machine Learning Models, Empirical Risk Minimization, Parameter Estimation, Probabilistic Modelling and Inference, Directed Graphical Models, Model Selection.

UNIT IV MACHINE LEARNING MODELS AND APPLICATIONS   6

Linear Regression, Parameter Estimation, Dimensionality Reduction with Principal Component Analysis, Maximum Variance Perspective, PCA in High Dimensions, Latent Variable Perspective, EM Algorithm, Latent-Variable, Classification with Support Vector Machines.

UNIT V DEEP LEARNING MODELS   6

Tensors from Machine Learning and Data Science, Deep Convolutional Neural Network Architectures for Image Classification, Latent Space and Generative Modelling, Autoencoders and Variational Autoencoders.

TOTAL NUMBER OF PERIODS INCLUDING LAB: 30

SAMPLE LIST OF EXPERIMENTS   **NUMBER OF PRACTICAL PERIODS:**   **30**

Use python for implementing the following.

1. Systems of Linear Equations Solver: Implement a program to solve systems of linear equations using methods like Gaussian elimination or LU decomposition.
2. Vector Space Visualization: Use a computational tool like MATLAB or Python with libraries such as NumPy and Matplotlib to visualize vector spaces, linear independence, basis, and rank.
3. Matrix Decomposition Analysis: Implement algorithms for matrix decomposition techniques such as eigen-decomposition, singular value decomposition (SVD), and Cholesky decomposition. Analyze their properties and computational complexities.
4. Gradient Computation: Develop code to compute gradients for univariate and multivariate functions using analytical methods and numerical approximation techniques like finite differences.
5. Optimization Algorithms: Implement gradient descent and its variants for unconstrained optimization problems. Study their convergence properties and compare their performance on benchmark functions.
6. Constrained Optimization with Lagrange Multipliers: Solve constrained optimization problems using the Lagrange multiplier method. Explore its applications in machine learning models and analyze trade-offs between constraints and objectives.
7. Probabilistic Modeling and Inference: Develop a probabilistic model for a given dataset and perform inference using methods like maximum likelihood estimation (MLE) or Bayesian inference. Implement algorithms for parameter estimation and compare their performance.
8. Support Vector Machine (SVM) Classifier: Implement a linear SVM classifier from scratch using optimization techniques like gradient descent or quadratic programming. Evaluate its performance on benchmark datasets and compare it with other classification methods.
9. Principal Component Analysis (PCA): Implement PCA for dimensionality reduction on datasets with high dimensions. Visualize the transformed data and analyze the explained variance ratio of principal components.
10. Deep Convolutional Neural Network (CNN) for Image Classification: Implement a CNN architecture using a deep learning framework like TensorFlow or PyTorch. Train the model on image classification tasks and evaluate its performance on standard datasets like CIFAR-10 or MNIST.

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   To understand fundamental linear algebra concepts, including vector spaces, matrices, eigenvalues, and eigenvectors.

CO2   To gain proficiency in advanced calculus techniques, including gradient-based optimization, to train and fine-tune learning models effectively for optimal performance.

CO3   Explore the role of probability and statistics in learning and understanding their significance in model training, uncertainty estimation, and probabilistic modelling.

CO4   To apply mathematical models through hands-on projects, implementing machine learning models.

CO5   To explore implementing deep learning models and analyse their performance using

mathematical tools.

TEXTBOOKS:

1. Eugene Charniak, "Introduction to Deep Learning," MIT Press, 2018.
2. Ivan Vasilev, Daniel Slater, Gianmario Spacagna, Peter Roelants, Valentino Zocca, "Python Deep Learning," Packt Publishing Ltd, 2019.

REFERENCES:

1. Ian Goodfellow, Yoshua Bengio, and Aaron Courville, " Deep Learning," MIT Press, 2017.
2. Josh Patterson, Adam Gibson, "Deep Learning: A Practitioner's Approach" O'Reilly Media, 2017.
3. Umberto Michelucci "Applied Deep Learning: A Case-based Approach to Understanding Deep Neural Networks" Apress, 2018.
4. Kevin P. Murphy, "Machine Learning: A Probabilistic Perspective" The MIT Press, 2012.
5. EthemAlpaydin,"Introduction to Machine Learning," MIT Press, Prentice Hall of India, Third Edition 2014.
6. Giancarlo Zaccone, Md. Rezaul Karim, Ahmed Menshawy, "Deep Learning with TensorFlow: Explore Neural Networks with Python" Packt Publisher, 2017.
7. Antonio Gulli, Sujit Pal, "Deep Learning with Keras" Packt Publishers, 2017.
8. Francois Chollet, "Deep Learning with Python," Manning Publications, 2017

Course Code	DEEP LEARNING	L	T	P	C
CSPE3XV43		2	0	2	3

COURSE OBJECTIVES:

1. To understand the basic ideas and principles of neural networks and concepts of deep learning.
2. To study Convolutional Neural Networks with image processing facilities like TensorFlow and Keras.
3. To study Recurrent Neural Networks with speech processing models
4. To study Deep Reinforcement Learning and the use of real-time applications.
5. To understand and implement deep learning architectures.

UNIT I FUNDAMENTALS OF DEEP LEARNING   **6**

Introduction to Neural Network – Feed Forward Neural Nets – Tensorflow - Deep Learning Fundamentals: Fundamental deep learning concepts, deep learning algorithms, and their types

UNIT II CONVOLUTIONAL NEURAL NETWORK   **6**

Convolutional Neural Networks – Filters – Strides and Padding – The structure of a convolutional network – Improving the performance of CNNs - Multilevel Convolution – Computer Vision with Convolutional Networks – Advanced Computer Vision

UNIT III RECURRENT NEURAL NETWORK   **6**

Recurrent Neural Networks: Introduction – Recursive Neural Networks – Bidirectional RNNs – Deep Recurrent Networks - Complete Autoencoder, Regularized Autoencoder, Stochastic Encoders and Decoders, Contractive Encoders – Language Modelling – Sequence to sequence learning – Speech Recognition

UNIT IV DEEP REINFORCEMENT LEARNING   **6**

Reinforcement Learning Theory – Markov Decision process – Monte Carlo methods – Temporal Difference methods – Value functions – Q learning – Deep Q-learning – Policy gradient methods – Model-based methods -Actor-Critic Methods

UNIT V DEEP LEARNING IN AUTONOMOUS VEHICLES   **6**

Autonomous Vehicles Introduction – Imitation driving policy – Driving policy with ChaufferNet – DL in Cloud.

TOTAL NUMBER OF PERIODS INCLUDING LAB: 60

SAMPLE LIST OF EXPERIMENTS   **NUMBER OF PRACTICAL PERIODS: 30**

11. Implement a feedforward neural network using TensorFlow to classify handwritten digits from the MNIST dataset.

12. Design a convolutional neural network with appropriate filters and padding to classify images from the CIFAR-10 dataset.

13. Compare the performance of different stride values in convolutional layers on a given image recognition task.

14. Explore the impact of multilevel convolutions on improving the accuracy of a CNN for object detection in computer vision.

15. Build a recurrent neural network model to generate text sequences and analyze its performance in language modelling.
16. Develop a bidirectional RNN architecture for sentiment analysis on movie reviews dataset and compare it with a unidirectional RNN.
17. Implement a deep reinforcement learning agent using Q-learning to solve a simple grid world problem.
18. Evaluate the performance of Deep Q-learning algorithm on the CartPole environment in OpenAI Gym.
19. Design an autonomous driving policy using imitation learning and assess its performance in a simulated environment.
20. Investigate the feasibility of deploying a deep learning model for autonomous driving on cloud infrastructure, considering latency and scalability aspects.

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Understanding the basic concepts of deep learning.

CO2   Emphasizing knowledge of Convolutional Neural Networks and applying CNN to its variants for suitable applications.

CO3   Understanding Recurrent Neural Networks to apply autoencoders and generative models for suitable applications.

CO4   Understanding deep reinforcement learning

CO5   Analysing the key computations underlying deep learning and using them to build and train deep neural networks for various tasks.

TEXTBOOKS:

1. Eugene Charniak, "Introduction to Deep Learning," MIT Press, 2018.
2. Ivan Vasilev, Daniel Slater, Gianmario Spacagna, Peter Roelants, Valentino Zocca, "Python Deep Learning," Packt Publishing Ltd, 2019.

REFERENCES:

1. Ian Goodfellow, Yoshua Bengio, and Aaron Courville, " Deep Learning," MIT Press, 2017.
2. Josh Patterson, Adam Gibson, "Deep Learning: A Practitioner's Approach" O'Reilly Media, 2017.
3. Umberto Michelucci "Applied Deep Learning: A Case-based Approach to Understanding Deep Neural Networks" Apress, 2018.
4. Kevin P. Murphy, "Machine Learning: A Probabilistic Perspective" The MIT Press, 2012.
5. EthemAlpaydin,"Introduction to Machine Learning," MIT Press, Prentice Hall of India, Third Edition 2014.
6. Giancarlo Zaccone, Md. Rezaul Karim, Ahmed Menshawy, "Deep Learning with TensorFlow: Explore Neural Networks with Python" Packt Publisher, 2017.
7. Antonio Gulli, Sujit Pal, "Deep Learning with Keras" Packt Publishers, 2017.
8. Francois Chollet, "Deep Learning with Python," Manning Publications, 2017

Course Code	NATURAL LANGUAGE PROCESSING	L	T	P	C
CSPE3XV44		2	0	2	3

COURSE OBJECTIVES:

1. To learn the mathematical foundations and basics of Natural Language Processing.
2. To understand the text data processing technologies for processing text data.
3. To understand the role of Information Retrieval and Information Extraction in Text Analytics.
4. To acquire knowledge of text data analytics using language models.
5. To learn about NLP Tools and real-time examples of NLP.

UNIT I INTRODUCTION TO NATURAL LANGUAGE PROCESSING **6**

Natural Language Processing – Linguistic Background – Mathematical Foundations – Morphological Analysis – Tokenization – Stemming – Lemmatization – Boundary Determination.

UNIT II TEXT DATA ANALYSIS **6**

Reading unstructured data – Representing text data – Part of speech tagging – Syntactic representation – Text similarity – WordNet-based similarity – Shallow parsing – Semantic representation.

UNIT III INFORMATION RETRIEVAL AND EXTRACTION **6**

Information Retrieval: Design features of Information Retrieval Systems-Classical, Nonclassical, Alternative Models of Information Retrieval – Information extraction – Named Entity Recognition – Relation Identification - Template filling.

UNIT IV LANGUAGE MODELLING **6**

Language model – Probabilistic Models – n-gram language models- Hidden Markov Model- Topic Modelling - Graph Models -Feature Selection and classifiers -Rule-based Classifiers - Maximum entropy classifier – Clustering-Word and Phrase-based Clustering.

UNIT V NLP TOOLS AND APPLICATIONS **6**

Tools: Natural Language Toolkit, Apache OpenNLP. Applications of Text Analytics – Applications in social media - Life science - Legal Text – Visualization - Case studies.

TOTAL NUMBER OF PERIODS INCLUDING LAB: 60

SAMPLE LIST OF EXPERIMENTS   NUMBER OF PRACTICAL PERIODS: 30

1. Implement tokenization and compare the effectiveness of stemming versus lemmatization in improving text preprocessing for sentiment analysis.
2. Develop a part-of-speech tagging system using NLTK and evaluate its accuracy on a corpus of news articles.
3. Explore various text similarity metrics, including WordNet-based similarity, for clustering news headlines into topics.
4. Build an information retrieval system using classical and nonclassical models and compare their performance on a dataset of scientific papers.
5. Implement a named entity recognition model using Apache OpenNLP and assess its

accuracy on legal text documents.

6. Investigate different approaches for relation identification in biomedical texts and evaluate their precision and recall.
7. Construct a language model using n-gram models and compare its performance with a hidden Markov model on a corpus of tweets.
8. Apply topic modeling techniques to extract themes from a collection of customer reviews and visualize the results using t-SNE.
9. Develop a rule-based classifier to categorize legal documents into different types and measure its accuracy against a maximum entropy classifier.
10. Utilize word and phrase-based clustering algorithms to identify patterns in social media conversations and analyze their implications for marketing strategies.

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1	Understand the mathematical foundations and basics of Natural Language Processing.
CO2	Process text data at the syntactic and semantic level.
CO3	Extract key information from text data.
CO4	Analyze text content to provide predictions related to a specific domain using language processing.
CO5	Design an innovative application using NLP components.

TEXTBOOKS:

1. Christopher D. Manning and Hinrich Schutze, "Foundations of Statistical Natural Language Processing," MIT Press, 1999;
2. Steven Struhl, "Practical Text Analytics: Interpreting Text and Unstructured Data
3. for Business Intelligence", Kogan Page, 2015.
4. Steven Bird, Ewan Klein and Edward Loper, "Natural Language Processing with
5. Python", 1st Edition, O'Reilly Media, 2009.

REFERENCES:

1. Matthew A. Russell, "Mining the Social Web," O'Reilly Media, 2013;
2. Daniel Jurafsky and James H. Martin "Speech and Language Processing," 3rd edition, Prentice Hall, 2009.
3. NitinIndurkhya, Fred J. Damerau "Handbook of Natural Language Processing," Second Edition, CRC Press, 2010.

Course Code	COMPUTER VISION AND IMAGE PROCESSING	L	T	P	C
CSPE3XV45		2	0	2	3

COURSE OBJECTIVES:

1. To understand the basics of image processing techniques for computer vision.
2. To learn the techniques used for image pre-processing.
3. To discuss the various object detection techniques.
4. To understand the various Object recognition mechanisms.
5. To elaborate on the video analytics techniques.

UNIT I INTRODUCTION 6

Computer Vision – Image representation and image analysis tasks - Image representations – digitization – properties – color images – Data structures for Image Analysis - Levels of image data representation - Traditional and Hierarchical image data structures.

UNIT II IMAGE PRE-PROCESSING 6

Local pre-processing - Image smoothing - Edge detectors - Zero-crossings of the second derivative - Scale in image processing - Canny edge detection - Parametric edge models - Edges in multi-speralct images - Local pre-processing in the frequency domain – Line detection by local pre-processing operators - Image restoration.

UNIT III OBJECT DETECTION USING MACHINE LEARNING 6

Object detection– Object detection methods – Deep Learning framework for Object detection– bounding box approach-Intersection over Union (IoU) –Deep Learning.

Architectures-R-CNN-Faster R-CNN-You Only Look Once(YOLO)-Salient features-Loss Functions-YOLO architectures

UNIT IV FACE RECOGNITION AND GESTURE RECOGNITION 6

Face Recognition-Introduction-Applications of FaceRecognition-Process of Face Recognition- DeepFace solution by Facebook-FaceNet for Face Recognition-Implementation using FaceNet- Gesture Recognition.

6

UNIT V VIDEO ANALYTICS

Video Processing – use cases of video analytics-Vanishing Gradient and exploding gradient problem-Rest Net architecture-RestNet and skip connections-Inception Network-Google Net architecture-Improvement in Inception v2-Video analytics-Rest Net and Inception v3.

TOTAL NUMBER OF PERIODS INCLUDING LAB: 60

SAMPLE LIST OF EXPERIMENTS   NUMBER OF PRACTICAL PERIODS: 30

1. Write a program that computes the T-pyramid of an image.
2. Write a program that derives the quad tree representation of an image using the homogeneity criterion of equal intensity
3. Develop programs for the following geometric transforms:

    (a) Rotation (b) Change of scale

4. Skewing (d) Affine transform calculated from three pairs of corresponding points
   (e) Bilinear transform calculated from four pairs of corresponding points.

5. Develop a program to implement Object Detection and Recognition

6. Develop a program for motion analysis using moving edges, and apply it to your image sequences.

7. Develop a program for Facial Detection and Recognition.

8. Write a program for event detection in video surveillance system

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Understand the basics of image processing techniques for computer vision and video analysis.
CO2   Explain the techniques used for image pre-processing.
CO3   Develop various object detection techniques.
CO4   Understand the various face recognition mechanisms.
CO5   Elaborate on deep learning-based video analytics.

TEXTBOOKS:

1. Milan Sonka, Vaclav Hlavac, Roger Boyle, "Image Processing, Analysis, and Machine Vision", 4nd edition, Thomson Learning, 2013.

2. Vaibhav Verdhan,(2021, Computer Vision Using Deep Learning Neural

   Network Architectures with Python and Keras,Apress 2021(UNIT-III,IV and V)

REFERENCES:

1. Richard Szeliski, "Computer Vision: Algorithms and Applications", Springer Verlag London Limited,2011.

2. Caifeng Shan, FatihPorikli, Tao Xiang, Shaogang Gong, "Video Analytics for Business Intelligence", Springer, 2012.

3. D. A. Forsyth, J. Ponce, "Computer Vision: A Modern Approach", Pearson Education, 2003.

4. E. R. Davies, (2012), "Computer & Machine Vision", Fourth Edition, Academic Press.

Course Code	REINFORCEMENT LEARNING	L	T	P	C
CSPE3XV46		2	0	2	3

COURSE OBJECTIVES:

1. Explore the historical development and interdisciplinary connections of Reinforcement Learning.
2. Gain a deep understanding of Markov Decision Processes (MDPs)
3. Focusing on iterative policy evaluation and iteration, and understanding the convergence properties.
4. Understand Monte Carlo methods for model-free prediction and control. application in reinforcement learning tasks.
5. Familiarize with function approximation methods and their applications in reinforcement learning.

UNIT I INTRODUCTION 6

Introduction- Origin and history of Reinforcement Learning research. Its connections with other related fields and with different branches of machine learning.

Probability Primer - Brush up of Probability concepts - Axioms of probability, concepts of random variables, PMF, PDFs, CDFs, Expectation. Concepts of joint and multiple random variables, joint, conditional and marginal distributions. Correlation and independence.

UNIT II MARKOV DECISION PROCESS 6

Introduction to RL terminology, Markov property, Markov chains, Markov reward process (MRP). Introduction to and proof of Bellman equations for MRPs along with proof of existence of solution to Bellman equations in MRP. Introduction to Markov decision process (MDP), state and action value functions, Bellman expectation equations, optimality of value functions and policies, Bellman optimality equations.

UNIT III PREDICTION AND CONTROL BY DYNAMIC PROGRAMING 6

Overview of dynamic programming for MDP, definition and formulation of planning in MDPs, principle of optimality, iterative policy evaluation, policy iteration, value iteration, Banach fixed point theorem, proof of contraction mapping property of Bellman expectation and optimality operators, proof of convergence of policy evaluation and value iteration algorithms, DP extensions.

UNIT IV MONTE CARLO METHODS FOR MODEL FREE PREDICTION AND CONTROL 6

Overview of Monte Carlo methods for model-free RL, First visit and every visit Monte Carlo, Monte Carlo control, On-policy and off-policy learning, Importance sampling. **TD Methods** Incremental Monte Carlo Methods for Model Free Prediction, Overview TD(0), TD(1) and TD(λ), k-step estimators, unified view of DP, MC, and TD evaluation methods, TD Control methods - SARSA, Q-Learning and their variants.

UNIT V FUNCTION APPROXIMATION METHODS 6

Getting started with the function approximation methods, Revisiting risk minimization, gradient descent from Machine Learning, Gradient MC and Semi-gradient TD(0) algorithms, Eligibility trace for function approximation, Afterstates, Control with function approximation, Least squares, Experience replay in deep Q-Networks.

Policy Gradients - Getting started with policy gradient methods, Log-derivative trick, Naive REINFORCE algorithm, bias and variance in Reinforcement Learning, Reducing variance in policy gradient estimates, baselines, advantage function, actor-critic methods.

TOTAL NUMBER OF PERIODS INCLUDING LAB:60

SAMPLE LIST OF EXPERIMENTS **NUMBER OF PRACTICAL PERIODS: 30**

1. Simulation of a Markov Chain: Simulate a simple Markov chain to demonstrate its properties and transitions between states.
2. Bellman Equation Implementation: Implement the Bellman equation for a Markov reward process in a simple environment to understand its application in reinforcement learning.
3. Policy Evaluation with Dynamic Programming: Implement policy evaluation using iterative methods like policy iteration or value iteration for a simple Markov decision process.
4. Monte Carlo Prediction: Implement first-visit Monte Carlo prediction to estimate state values in a grid world environment without a model.
5. Q-Learning Implementation: Implement the Q-learning algorithm for solving a simple grid world problem, demonstrating the exploration-exploitation trade-off.
6. Function Approximation with Linear Regression: Implement linear regression as a function approximation method in reinforcement learning to approximate state-action values.
7. Actor-Critic Method Implementation: Implement an actor-critic algorithm to learn policies and value functions concurrently, demonstrating the advantage of bootstrapping.
8. Gradient Descent in Function Approximation: Implement gradient descent for updating parameters in a function approximation method like neural networks for Q-value estimation.
9. Experience Replay in Deep Q-Networks: Implement experience replay in a deep Q-network (DQN) to improve learning efficiency and stability.
10. Policy Gradient Method Implementation: Implement a policy gradient method like REINFORCE to learn a policy in a simple environment, analyzing bias and variance in the estimates.

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1 Attain comprehensive understanding of RL's historical evolution and interdisciplinary connections, alongside fundamental Probability concepts.
CO2 Achieve deep comprehension of MDPs, emphasizing terminology, properties, and Bellman equations for optimal decision-making.
CO3 Master Dynamic Programming techniques for MDP prediction and control tasks, understanding convergence properties.
CO4 Gain thorough understanding of Monte Carlo methods for model-free RL, proficiently implementing First visit and every visit techniques.
CO5 Familiarize with function approximation methods, gradient descent, eligibility traces, experience replay, policy gradient methods, and actor-critic architectures in RL

applications.

TEXTBOOKS:

1. Richard S. Sutton and Andrew G. Barto, "Reinforcement Learning: An Introduction",
   , 2nd Edition.
2. Sutton, R. S., & Barto, A. G. (2018). Reinforcement learning: An introduction (2nd ed.). MIT Press.

REFERENCES:

1. Leon-Garcia, A. (2008). Probability, statistics, and random processes for electrical engineering. Prentice Hall.
2. Murphy, K. P. (2012). Machine learning: A probabilistic perspective. MIT Press.

Course Code	BIG DATA ANALYTICS	L	T	P	C
CSPE3XV47		2	0	2	3

COURSE OBJECTIVES:

1. To understand big data
2. To learn and use NoSQL big data management.
3. To learn MapReduce analytics using Hadoop and related tools.
4. To work with map-reduce applications
5. To understand the usage of Hadoop-related tools for Big Data Analytics.

UNIT I UNDERSTANDING BIG DATA   **6**

Introduction to big data – convergence of key trends – unstructured data – industry examples of big data – web analytics – big data applications– big data technologies – introduction to Hadoop – open source technologies – cloud and big data – mobile business intelligence – Crowd sourcing analytics – inter and trans firewall analytics.

UNIT II NOSQL DATA MANAGEMENT   **6**

Introduction to NoSQL – aggregate data models – key-value and document data models – relationships – graph databases – schemaless databases – materialized views – distribution models – master-slave replication – consistency - Cassandra – Cassandra data model –
Cassandra examples – Cassandra clients

UNIT III MAP REDUCE APPLICATIONS   **6**

MapReduce workflows – unit tests with MRUnit – test data and local tests – anatomy of MapReduce job run – classic Map-reduce – YARN – failures in classic Map-reduce and YARN
– job scheduling – shuffle and sort – task execution – MapReduce types – input formats – output formats.

UNIT IV BASICS OF HADOOP   **6**

Data format – analyzing data with Hadoop – scaling out – Hadoop streaming – Hadoop pipes – design of Hadoop distributed file system (HDFS) – HDFS concepts – Java interface – data flow – Hadoop I/O – data integrity – compression – serialization – Avro – file-based data structures - Cassandra – Hadoop integration.

UNIT V HADOOP RELATED TOOLS   **6**

Hbase – data model and implementations – Hbase clients – Hbase examples – praxis. Pig – Grunt – pig data model – Pig Latin – developing and testing Pig Latin scripts.
Hive – data types and file formats – HiveQL data definition – HiveQL data manipulation –
HiveQL queries.

TOTAL NUMBER OF PERIODS INCLUDING LAB: 60

SAMPLE LIST OF EXPERIMENTS   NUMBER OF PRACTICAL PERIODS: 30

1. Downloading and installing Hadoop; Understanding different Hadoop modes. Startup scripts, Configuration files.
2. Hadoop Implementation of file management tasks, such as Adding files and directories, retrieving files and Deleting files
3. Implement of Matrix Multiplication with Hadoop Map Reduce
4. Run a basic Word Count Map Reduce program to understand Map Reduce Paradigm.
5. Installation of Hive along with practice examples.
7. Installation of HBase, Installing thrift along with Practice examples
8. Practice importing and exporting data from various databases.

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Describe big data and use cases from selected business domains.
CO2   Explain NoSQL big data management.
CO3   Install, configure, and run Hadoop and HDFS.
CO4   Perform map-reduce analytics using Hadoop.
CO5   Use Hadoop-related tools such as HBase, Cassandra, Pig, and Hive for big data analytics.

TEXTBOOKS:

1. Michael Minelli, Michelle Chambers, and AmbigaDhiraj, "Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's   Businesses", Wiley, 2013.
2. Eric Sammer, "Hadoop Operations", O'Reilley, 2012.
3. Sadalage, Pramod J. "NoSQL distilled", 2013

REFERENCES:

1. E. Capriolo, D. Wampler, and J. Rutherglen, "Programming Hive", O'Reilley, 2012.
2. Lars George, "HBase: The Definitive Guide", O'Reilley, 2011.
3. Eben Hewitt, "Cassandra: The Definitive Guide", O'Reilley, 2010.
4. Alan Gates, "Programming Pig", O'Reilley, 2011.

Course Code	**GENERATIVE ARTIFICIAL INTELLIGENCE**	**L**	**T**	**P**	**C**
CSPE3XV48		2	0	2	3

COURSE OBJECTIVES:

1. Understand the principles and theory behind generative AI.
2. Gain practical experience in developing generative AI models.
3. Analyse and evaluate the ethical and societal implications of generative AI.
4. Apply generative AI techniques to real-world problems and domains.
5. Keep up-to-date with the latest developments and trends in the field of generative AI.

Course Description   **6**

This course provides an introduction to the theory and practical applications of Generative Artificial Intelligence. Students will learn the fundamental concepts and techniques related to generative models and gain hands-on experience with creating and using generative AI systems.

Prerequisites

- Basic knowledge of machine learning and deep learning.
- Familiarity with a programming language (e.g., Python).

UNIT I Introduction to Generative AI   **9**

Overview of Generative AI and its applications – Difference between generative and discriminative models – Historical perspective and key milestones – Ethical and societal implications – Introduction to Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), Long Short-Term Memory Networks (LSTMs), Transformers, and Large Language Models (LLMs).

UNIT II Probability and Statistics for Generative AI   **9**

Probability distributions and their role in generative models – Maximum Likelihood Estimation (MLE) – Bayesian Inference and Maximum a Posteriori (MAP) estimation – Generative models as probabilistic models – Application of statistical methods in CNNs, RNNs, and LSTMs.

UNIT III Generative Models   **9**

Introduction to Autoencoders – Variational Autoencoders (VAE) – Generative Adversarial Networks (GAN) – Flow-based models – Practical implementation and hands-on exercises (using TensorFlow, PyTorch, Jupyter Notebook, Keras, etc) – Overview of Transformer models and their role in Generative AI – Practical implementation of Transformers and LLMs (e.g., BERT, GPT).

UNIT IV Applications of Generative AI   **9**

mage generation and manipulation – Text generation and natural language processing – Anomaly detection and data augmentation – Style transfer and artistic applications – Real-world use cases (Art & Design, Medical Imaging, Content creation, Chatbots, Virtual Assistants, Cybersecurity, etc.) and industry examples – Advanced applications of CNNs in image processing – Applications of RNNs and LSTMs in sequential data –

Guest Lectures by Industry Experts and Researchers.

UNIT V Evaluation and Ethical Considerations   **9**

Metrics for evaluating generative models (e.g., Inception Score, FID) – Ethical concerns in generative AI, including bias and fairness – Privacy and security considerations – Future trends and emerging technologies in Generative AI – Ethical implications of using LLMs – Case studies on ethical challenges in deploying generative models.

Course Format

Lectures and discussions, Hands-on coding exercises and projects, Guest lectures by industry Experts, Group discussions and presentations, Online resources and tutorials

Assessments & Grading

Quizzes / Assignments, Project, 3 Internal Assessments, Final Examination

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

CO1: Understand the fundamental principles and applications of Generative Artificial Intelligence and distinguish between generative and discriminative models.
CO2: Apply probability and statistics concepts to analyze and model data, with a focus on their role in generative models.
CO3: Implement and evaluate various generative models, including Autoencoders, Variational Autoencoders (VAE), and Generative Adversarial Networks (GANs).
CO4: Apply generative AI techniques to solve real-world problems, including image generation, text generation, and artistic applications.
CO5: Evaluate generative models using appropriate metrics and critically analyze the ethical implications, privacy concerns, and societal impact of generative AI technologies.

TEXT BOOKS:

1. David Foster, "Generative Deep Learning", Second Edition, O'Reilly Media, 2023.
2. Joseph Babcock and Raghav Bali, "Generative AI with Python and TensorFlow 2", Packt Publishing, 2021
3. Denis Rothman, "Transformers for Natural Language Processing", Second Edition, Packt Publishing, 2022.

REFERENCES:

4. Ian Goodfellow, Yoshua Bengio, Aaron Courville, "Deep Learning", The MIT Press, 2016.
5. Alberto Chierici, "The Ethics of AI", New Degree Press, 2021.
6. Jacob Emerson, "Ripples of Generative AI", IngramSpark, 2023.
7. Francois Chollet, "Deep Learning with Python", Second Edition, Manning, 2021.
8. Sebastian Raschka, Yuxi (Hayden) Liu, Vahid Mirjalili, "Machine Learning with PyTorch and Scikit-Learn", Packt Publishing, 2022.

YouTube Resources:

1. **3Blue1Brown** - While not specific to generative AI, this channel provides fantastic visual explanations of mathematics and concepts that are fundamental to AI and machine learning.
2. **sentdex** - This channel focuses on machine learning and AI using Python. It includes tutorials on various topics, including generative models.

Vertical V

CLOUD COMPUTING

Course Code	CLOUD TECHNOLOGIES	L	T	P	C
CSPE3XV51		2	0	2	3

COURSE OBJECTIVES:

1. To understand the principles of cloud architecture, models and infrastructure.
2. To understand the concepts of virtualization and virtual machines.
3. To gain knowledge about virtualization Infrastructure.
4. To explore and experiment with various Cloud deployment environments.
5. To learn about the security issues in the cloud environment.

UNIT I CLOUD ARCHITECTURE MODELS AND INFRASTRUCTURE   **6**

Cloud Architecture: System Models for Distributed and Cloud Computing – NIST Cloud Computing Reference Architecture – Cloud deployment models – Cloud service models; Cloud Infrastructure: Architectural Design of Compute and Storage Clouds – Design Challenges.

UNIT II VIRTUALIZATION BASICS   **6**

Virtual Machine Basics – Taxonomy of Virtual Machines – Hypervisor – Key Concepts – Virtualization structure – Implementation levels of virtualization – Virtualization Types: Full Virtualization – Para Virtualization – Hardware Virtualization – Virtualization of CPU, Memory and I/O devices.

UNIT III VIRTUALIZATION INFRASTRUCTURE AND DOCKER   **6**

Desktop Virtualization – Network Virtualization – Storage Virtualization – System-level of Operating Virtualization – Application Virtualization – Virtual clusters and Resource Management – Containers vs. Virtual Machines – Introduction to Docker – Docker Components – Docker Container – Docker Images and Repositories.

UNIT IV CLOUD DEPLOYMENT ENVIRONMENT   **6**

Google App Engine – Amazon AWS – Microsoft Azure; Cloud Software Environments – Eucalyptus – OpenStack.

UNIT V CLOUD SECURITY   **6**

Virtualization System-Specific Attacks: Guest hopping – VM migration attack – hyperjacking. Data Security and Storage; Identity and Access Management (IAM) - IAM Challenges - IAM Architecture and Practice.

TOTAL NUMBER OF PERIODS INCLUDING LAB: 60

SAMPLE LIST OF EXPERIMENTS   NUMBER OF PRACTICAL PERIODS: 30

1. Install Virtualbox/VMware/ Equivalent open source cloud Workstation with different flavours of Linux or Windows OS on top of windows 8 and above.
2. Install a C compiler in the virtual machine created using a virtual box and execute Simple Programs

3. Install Google App Engine. Create a hello world app and other simple web applications using python/java.
4. Use the GAE launcher to launch the web applications.
5. Simulate a cloud scenario using CloudSim and run a scheduling algorithm that is not present in CloudSim.
6. Find a procedure to transfer the files from one virtual machine to another virtual machine.
7. Install Hadoop single node cluster and run simple applications like wordcount.
8. Creating and Executing Your First Container Using Docker.
9. Run a Container from Docker Hub

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Understand the design challenges in the cloud.

CO2   Apply the concept of virtualization and its types.

CO3   Experiment with virtualization of hardware resources and Docker.

CO4   Develop and deploy services on the cloud and set up a cloud environment.

CO5   Explain security challenges in the cloud environment.

TEXTBOOKS:

1. Kai Hwang, Geoffrey C Fox, Jack G Dongarra, "Distributed and Cloud Computing, From Parallel Processing to the Internet of Things", Morgan Kaufmann Publishers, 2012.
2. James Turnbull, "The Docker Book", O'Reilly Publishers, 2014.
3. Krutz, R. L., Vines, R. D, "Cloud security. A Comprehensive Guide to Secure Cloud Computing", Wiley Publishing, 2010.

REFERENCES:

1. James E. Smith, Ravi Nair, "Virtual Machines: Versatile Platforms for Systems and Processes", Elsevier/Morgan Kaufmann, 2005.
2. Tim Mather, Subra Kumaraswamy, and Shahed Latif, "Cloud Security and Privacy: an enterprise perspective on risks and compliance", O'Reilly Media, Inc., 2009.

Course Code	VIRTUALIZATION	L	T	P	C
CSPE3XV52		2	0	2	3

COURSE OBJECTIVES:

1. To Learn the basics and types of Virtualization
2. To understand the Hypervisors and its types
3. To Explore the Virtualization Solutions
4. To Experiment the virtualization platforms

UNIT I INTRODUCTION TO VIRTUALIZATION   **6**

Virtualization and cloud computing - Need of virtualization – cost, administration, fast deployment, reduce infrastructure cost – limitations- Types of hardware virtualization: Full virtualization - partial virtualization - Paravirtualization-Types of Hypervisors

UNIT II SERVER AND DESKTOP VIRTUALIZATION   **6**

Virtual machine basics- Types of virtual machines- Understanding Server Virtualization- types of server virtualization- Business Cases for Server Virtualization – Uses of Virtual Server Consolidation – Selecting Server Virtualization Platform-Desktop Virtualization-Types of Desktop Virtualization

UNIT III NETWORK VIRTUALIZATION   **6**

Introduction to Network Virtualization-Advantages- Functions-Tools for Network Virtualization- VLAN-WAN Architecture-WAN Virtualization

UNIT IV STORAGE VIRTUALIZATION   **6**

Memory Virtualization-Types of Storage Virtualization-Block, File-Address space Remapping-Risks of Storage Virtualization-SAN-NAS-RAID

UNIT V VIRTUALIZATION TOOLS   **6**

VMWare-Amazon AWS-Microsoft HyperV- Oracle VM Virtual Box - IBM PowerVM- Google Virtualization- Case study.

TOTAL NUMBER OF PERIODS INCLUDING LAB: 60

SAMPLE LIST OF EXPERIMENTS   **NUMBER OF PRACTICAL PERIODS: 30**

1.Create type 2 virtualization in VMWARE or any equivalent Open-Source Tool. Allocate memory and storage space as per requirement. Install Guest OS on that VMWARE.

2. a.Shrink and extend virtual disk
   b.Create, Manage, Configure and schedule snapshots
   c.Create Spanned, Mirrored and Striped volume
   d.Create RAID 5 volume

3. a.Desktop Virtualization using VNC
   b.Desktop Virtualization using Chrome Remote Desktop

4.Create type 2 virtualization on ESXI 6.

5 server 5.Create a VLAN in CISCO packet tracer

6.Install KVM in Linux

7.Create Nested Virtual Machine(VM under another VM)

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Analyse the virtualization concepts and Hypervisor
CO2   Apply the Virtualization for real-world applications
CO3   Install & Configure the different VM platforms
CO4   Experiment with the VM with various software

TEXT CUM REFERENCE BOOKS:

1.Cloud computing a practical approach - Anthony T.Velte , Toby J. Velte Robert Elsenpeter, TATA McGraw- Hill , New Delhi – 2010

2.Cloud Computing (Principles and Paradigms), Edited by Rajkumar Buyya, James Broberg, Andrzej Goscinski, John Wiley & Sons, Inc. 2011

3.David Marshall, Wade A. Reynolds, Advanced Server Virtualization: VMware and Microsoft Platform in the Virtual Data Center, Auerbach

4.Chris Wolf, Erick M. Halter, "Virtualization: From the Desktop to the Enterprise", APress, 2005.

5.James E. Smith, Ravi Nair, "Virtual Machines: Versatile Platforms for Systems and Processes", Elsevier/Morgan Kaufmann, 2005.

6.David Marshall, Wade A. Reynolds, "Advanced Server Virtualization: VMware and Microsoft Platform in the Virtual Data Center", Auerbach Publications, 2006.

Course Code	CLOUD ARCHITECTURES	L	T	P	C
CSPE3XV53		2	0	2	3

COURSE OBJECTIVES:

1. Understand the Fundamentals of Cloud Computing
2. Explore Key Concepts in Cloud Architectures
3. Evaluate Cloud Service Providers and Deployment Models
4. Apply Design Principles for Building Cloud Architectures
5. Analyze Architectural Patterns and Networking Strategies in the Cloud

UNIT I FOUNDATIONS OF CLOUD COMPUTING AND ARCHITECTURAL CONCEPTS   6

Introduction to Cloud Computing – Definition and Characteristics, Evolution of Cloud Computing, Cloud Service Models (IaaS, PaaS, SaaS), Deployment Models (Public Cloud, Private Cloud, Hybrid Cloud). Key Concepts in Cloud Architectures – Virtualization, Scalability and Elasticity, Reliability and Availability, Security in the Cloud, Compliance and Governance.

UNIT II CLOUD SERVICE PROVIDERS AND ARCHITECTURAL DESIGN PRINCIPLES   6

Cloud Service Providers – Overview of Major Cloud Providers (AWS, Azure, Google Cloud, etc.), Comparative Analysis of Cloud Providers, Choosing the Right Cloud Service Provider. Design Principles for Cloud Architectures – Microservices Architecture, Serverless Computing, Containerization (Docker, Kubernetes), Decoupling and Asynchronous Communication.

UNIT III CLOUD ARCHITECTURE PATTERNS AND NETWORKING   6

Architectural Patterns in the Cloud – Multi-Tier Applications, Event-Driven Architectures, Big Data Architectures, IoT (Internet of Things) Architectures. Cloud Networking and Connectivity - Virtual Private Clouds, Content Delivery Networks (CDN), Hybrid Networking, Network Security in the Cloud.

UNIT IV CLOUD DATA MANAGEMENT AND PERFORMANCE OPTIMIZATION   6

Data Management in the Cloud – Cloud Databases (SQL and NoSQL), Data Storage Services, Data Backup and Recovery, Data Transfer and Migration Strategies. Performance Optimization and Monitoring – Resource Scaling and Auto-Scaling, Performance Monitoring and Logging, Cost Optimization Strategies, Troubleshooting and Debugging in the Cloud.

UNIT V Real-world Applications and Future Trends   6

Case Studies and Real-world Implementations – Successful Cloud Migrations, Cloud-Native Applications, Challenges and Lessons Learned. Future Trends in Cloud Architectures - Edge Computing, Quantum Computing and Cloud, Emerging Technologies.

TOTAL NUMBER OF PERIODS INCLUDING LAB: 60

SAMPLE LIST OF EXPERIMENTS   NUMBER OF PRACTICAL PERIODS: 30

1. Deploy a virtual machine on AWS. Install a web server and share the public IP address for verification.
2. Set up auto-scaling for an Azure web app. Show how it responds to a simulated increase in user traffic.
3. Create an AWS VPC with security groups. Launch an EC2 instance with a secure

connection. Share access details.
4. Dockerize three microservices (e.g., Node.js, Python, Java). Show how they communicate using Docker Compose.
5. Write an AWS Lambda function in Python. Configure an S3 bucket to trigger the function on object creation.
6. Build an AWS-based big data pipeline using S3, Lambda, and Athena. Show data ingestion and query results.
7. Set up an Azure VNet with front-end and back-end subnets. Implement Azure CDN for a web application.
8. Use Google Cloud Monitoring to track the performance of a Compute Engine instance. Implement auto-scaling based on metrics.
9. Analyze Netflix's cloud architecture. Identify key decisions, challenges, and benefits of their migration to the cloud.

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Master the Core Concepts of Cloud Computing

CO2   Demonstrate Proficiency in Designing Secure and Scalable Cloud Architectures

CO3   Evaluate and Select Appropriate Cloud Service Providers

CO4   Apply Architectural Patterns to Real-world Cloud Solutions

CO5   Implement Efficient Networking and Data Management Strategies in Cloud Environments

CO6   Critically Analyze and Troubleshoot Cloud-based Systems

TEXTBOOKS:

1. "Cloud Computing: Concepts, Technology & Architecture" by Thomas Erl, Ricardo Puttini, Zaigham Mahmood
2. "Architecting the Cloud: Design Decisions for Cloud Computing Service Models" by Michael J. Kavis
3. "Cloud Native Patterns: Designing Change-tolerant Software" by Cornelia Davis

REFERENCES:

1. "The Art of Cloud Computing: Building Cloud-Based Applications and Infrastructure" by Rajkumar Buyya, Christian Vecchiola, S. Thamarai Selvi
2. "Cloud Computing: From Beginning to End" by Ray J. Rafaels
3. "Cloud Application Architectures: Building Applications and Infrastructure in the Cloud" by George Reese.

Course Code	CLOUD PLATFORM PROGRAMMING	L	T	P	C

CSPE3XV54		2	0	2	3

COURSE OBJECTIVES:

1. Ability to Develop Proficiency in Cloud Service Providers
2. Master Cloud Programming Languages and Tools
3. Implement Cloud-natve Applications
4. Ensure Cloud Application Security
5. Apply DevOps Practices for Cloud Development

UNIT I INTRODUCTION TO CLOUD COMPUTING AND CLOUD PLATFORMS　　**6**

Overview of Cloud Computing;　　Evolution of Cloud Computing, Cloud Service Models (IaaS, PaaS, SaaS), Cloud Deployment Models (Public, Private, Hybrid), Major Cloud Service Providers (AWS, Azure, Google Cloud), Setting up Cloud Accounts and Environments, Cloud Platform Services (Compute, Storage, Networking), Identity and Access Management (IAM), Overview of Pricing and Billing in the Cloud.

UNIT II PROGRAMMING LANGUAGES AND TOOLS FOR CLOUD　　**6**

Overview of Programming Languages for Cloud Development, Cloud SDKs and CLIs, Infrastructure as Code (IaC) Concepts and Tools (e.g., Terraform), Containerization and Orchestration (Docker, Kubernetes).

UNIT III CLOUD APPLICATION DEVELOPMENT　　**6**

Serverless Computing and Functions as a Service (FaaS), Microservices Architecture in the Cloud, Cloud-native Development Best Practices, Data Storage and Databases in the Cloud, Testing and Debugging Cloud Applications

UNIT IV CLOUD SECURITY AND COMPLIANCE　　**6**

Security Challenges in the Cloud, Encryption and Key Management, Compliance and Governance in the Cloud, Best Practices for Securing Cloud Applications, Incident Response and Recovery in the Cloud.

UNIT V ADVANCED TOPICS IN CLOUD PROGRAMMING　　**6**

DevOps and Continuous Integration/Continuous Deployment (CI/CD), Monitoring and Logging in the Cloud, Advanced Cloud Services (e.g., AI/ML, IoT), Performance Optimization and Scalability.

TOTAL NUMBER OF PERIODS INCLUDING LAB:　60

SAMPLE LIST OF EXPERIMENTS　　**NUMBER OF PRACTICAL PERIODS:　30**

1. Create an account on a major cloud platform such as AWS, Azure, Google Cloud, or others.
2. Explore the platform's dashboard and understand the basic services offered.
3. Launch a virtual machine (VM) instance on the cloud platform.
4. Deploy a containerized application using a container orchestration tool like Docker or Kubernetes.
5. Create a serverless function using platforms like AWS Lambda, Azure Functions, or Google Cloud Functions.
6. Trigger the function in response to an event (e.g., HTTP request, file upload, database change).
7. Use cloud storage services (e.g., Amazon S3, Azure Blob Storage) to store and retrieve

files.
8. Implement versioning, access control, and lifecycle policies for stored objects.
9. Set up a managed database instance (e.g., AWS RDS, Azure SQL Database).
10. Connect your application to the database and perform CRUD operations.
11. Configure a virtual network with subnets, security groups, and network ACLs.
12. Implement SSL/TLS for securing communication between components.
13. Set up monitoring for your cloud resources using tools like AWS CloudWatch, Azure Monitor, or Google Cloud Monitoring.
14. Configure logging to track and analyze application and infrastructure logs.
15. Implement a CI/CD pipeline using services like AWS CodePipeline, Azure DevOps, or Google Cloud Build.
16. Automate the build, test, and deployment processes for your application.
17. Configure auto-scaling policies for your application based on metrics like CPU utilization.
18. Set up a load balancer to distribute incoming traffic across multiple instances.
19. Define IAM roles and policies to control access to your cloud resources.
20. Implement federated identity and Single Sign-On (SSO) if supported by the cloud platform.

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Able to explain the fundamental concepts of cloud computing, including service models deployment models, and the evolution of cloud technologies.

CO2   Will gain practical experience in setting up and using a cloud platform

CO3   Will acquire proficiency in programming for the cloud, using relevant programming languages, SDKs, and tools.

CO4   Will demonstrate an Understanding of cloud security challenges and solutions, including encryption, identity and access management, compliance.

CO5   Able to design and implement cloud-native applications, incorporating advanced concepts such as serverless computing.

TEXTBOOKS:
1. "Cloud Computing: From Beginning to End" by Ray J. Rafaels, Publisher: Apress, Edition: 1st edition (2016)
2. "Cloud Application Architectures: Building Applications and Infrastructure in the Cloud" by George Reese, Publisher: O'Reilly Media, Edition: 1st edition (2009)

REFERENCES:

3. "Programming Amazon EC2" by Jurg van Vliet and Flavia Paganelli, Publisher: O'Reilly Media, Edition: 1st edition (2011)
4. "Kubernetes: Up and Running" by Kelsey Hightower, Brendan Burns, and Joe Beda, Publisher: O'Reilly Media, Edition: 1st edition (2017)

Course Code	**CLOUD SERVICES MANAGEMENT**	L	T	P	C
CSPE3XV55		2	0	2	3

COURSE OBJECTIVES:

1. Introduce Cloud Service Management terminology, definition & concepts.
2. Compare and contrast cloud service management with traditional IT service management.
3. Identify strategies to reduce risk and eliminate issues associated with adoption of cloud services.
4. Select appropriate structures for designing, deploying and running cloud-based services in a business environment.
5. Illustrate the benefits and drive the adoption of cloud-based services to solve real world problems.

UNIT I CLOUD SERVICE MANAGEMENT FUNDAMENTALS **6**

Cloud Ecosystem, The Essential Characteristics, Basics of Information Technology Service Management and Cloud Service Management, Service Perspectives, Cloud Service Models, Cloud Service Deployment Models

UNIT II CLOUD SERVICES STRATEGY **6**

Cloud Strategy Fundamentals, Cloud Strategy Management Framework, Cloud Policy, Key Driver for Adoption, Risk Management, IT Capacity and Utilization, Demand and Capacity matching, Demand Queueing, Change Management, Cloud Service Architecture

UNIT III CLOUD SERVICE MANAGEMENT **6**

Cloud Service Reference Model, Cloud Service LifeCycle, Basics of Cloud Service Design, Dealing with Legacy Systems and Services, Benchmarking of Cloud Services, Cloud Service Capacity Planning, Cloud Service Deployment and Migration, Cloud Marketplace, Cloud Service Operations Management

UNIT IV CLOUD SERVICE ECONOMICS **6**

Pricing models for Cloud Services, Freemium, Pay Per Reservation, Pay per User, Subscription based Charging, Procurement of Cloud-based Services, Capex vs Opex Shift, Cloud service Charging, Cloud Cost Models

UNIT V CLOUD SERVICE GOVERNANCE & VALUE **6**

IT Governance Definition, Cloud Governance Definition, Cl
oud Governance Framework, Cloud Governance Structure, Cloud Governance Considerations, Cloud Service Model Risk Matrix, Understanding Value of Cloud Services, Measuring the value of Cloud Services, Scorecard, Total Cost of Ownership

TOTAL NUMBER OF PERIODS INCLUDING LAB: 60

SAMPLE LIST OF EXPERIMENTS   **NUMBER OF PRACTICAL PERIODS:**   **30**

1. Create a Cloud Organization in AWS/Google Cloud/or any equivalent Open Source cloud softwares like Openstack, Eucalyptus, OpenNebula with Role-based access control

2. Create a Cost-model for a web application using various services and do Cost-benefit analysis
3. Create alerts for usage of Cloud resources
4. Create Billing alerts for your Cloud Organization
5. Compare Cloud cost for a simple web application across AWS, Azure and GCP and suggest the best one

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Exhibit cloud-design skills to build and automate business solutions using cloud technologies.

CO2   Possess Strong theoretical foundation leading to excellence and excitement towards adoption of cloud-based services.

CO3   Solve the real world problems using Cloud services and technologies.

TEXTBOOKS:

1. Cloud Service Management and Governance: Smart Service Management in Cloud Era by Enamul Haque, Enel Publications
2. Cloud Computing: Concepts, Technology & Architecture by Thomas Erl, Ricardo Puttini, Zaigham Mohammad 2013
3. Cloud Computing Design Patterns by Thomas Erl, Robert Cope, Amin Naserpour

REFERENCES:

1. Economics of Cloud Computing by Praveen Ayyappa, LAP Lambert Academic Publishing
2. Mastering Cloud Computing Foundations and Applications Programming Rajkumar Buyya, Christian Vechhiola, S. Thamarai Selvi

Course Code	STREAM PROCESSING	L	T	P	C
CSPE3XV56		2	0	2	3

COURSE OBJECTIVES:

1 Introduce Data Processing terminology, definition & concepts

2 Define different types of Data Processing

3 Explain the concepts of Real-time Data processing

4 Select appropriate structures for designing and running real-time data services in a business environment

5 Illustrate the benefits and drive the adoption of real-time data services to solve real world problems.

UNIT I FOUNDATIONS OF DATA SYSTEMS **6**

Introduction to Data Processing, Stages of Data processing, Data Analytics, Batch Processing, Stream processing, Data Migration, Transactional Data processing, Data Mining, Data Management Strategy, Storage, Processing, Integration, Analytics, Benefits of Data as a Service, Challenges

UNIT II REAL-TIME DATA PROCESSING **6**

Introduction to Big data, Big data infrastructure, Real-time Analytics, Near real-time solution, Lambda architecture, Kappa Architecture, Stream Processing, Understanding Data Streams, Message Broker, Stream Processor, Batch & Real-time ETL tools, Streaming Data Storage

UNIT III DATA MODELS AND QUERY LANGUAGES **6**

Relational Model, Document Model, Key-Value Pairs, NoSQL, Object-Relational Mismatch, Many- to-One and Many-to-Many Relationships, Network data models, Schema Flexibility, Structured Query Language, Data Locality for Queries, Declarative Queries, Graph Data models, Cypher Query Language, Graph Queries in SQL, The Semantic Web, CODASYL, SPARQL

6
UNIT IV EVENT PROCESSING WITH APACHE KAFKA

Apache Kafka, Kafka as Event Streaming platform, Events, Producers, Consumers, Topics, Partitions, Brokers, Kafka APIs, Admin API, Producer API, Consumer API, Kafka Streams API, Kafka Connect API

UNIT V REAL-TIME PROCESSING USING SPARK STREAMING **6**

Structured Streaming, Basic Concepts, Handling Event-time and Late Data, Fault-tolerant Semantics, Exactly-once Semantics, Creating Streaming Datasets, Schema Inference, Partitioning of Streaming datasets, Operations on Streaming Data, Selection, Aggregation, Projection, Watermarking, Window operations, Types of Time windows, Join Operations, Deduplication

TOTAL NUMBER OF PERIODS INCLUDING LAB: 60

SAMPLE LIST OF EXPERIMENTS **NUMBER OF PRACTICAL PERIODS: 30**

1. Install MongoDB
2. Design and Implement Simple application using MongoDB
3. Query the designed system using MongoDB

4. Create a Event Stream with Apache Kafka
5. Create a Real-time Stream processing application using Spark Streaming
6. Build a Micro-batch application
7. Real-time Fraud and Anomaly Detection,
8. Real-time personalization, Marketing, Advertising

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Understand the applicability and utility of different streaming algorithms.
CO2   Describe and apply current research trends in data-stream processing.
CO3   Analyze the suitability of stream mining algorithms for data stream systems.
CO4   Program and build stream processing systems, services and applications.
CO5   Solve problems in real-world applications that process data streams.

TEXTBOOKS:

1. Streaming Systems: The What, Where, When and How of Large-Scale Data Processing by Tyler Akidau, Slava Chemyak, Reuven Lax, O'Reilly publication
2. Designing Data-Intensive Applications by Martin Kleppmann, O'Reilly Media
3. Practical Real-time Data Processing and Analytics : Distributed Computing and Event Processing using Apache Spark, Flink, Storm and Kafka, Packt Publishing

REFERENCES:

1. https://spark.apache.org/docs/latest/streaming-programming-guide.html
2. Kafka.apache.org

Course Code	FOG AND EDGE COMPUTING	L	T	P	C
CSPE3XV57		2	0	2	3

COURSE OBJECTIVES:

1. To enhance real-time data processing and analytics at the network edge.
2. To optimize resource utilization and reduce latency in fog computing environments.
3. To improve scalability and flexibility for edge devices and applications.
4. To Enhance security and privacy for data processing at the edge.
5. To Facilitate seamless integration of fog and edge computing with cloud services.

UNIT I INTRODUCTION TO FOG AND EDGE COMPUTING   **6**

Definition and Fundamentals: Overview of Fog and Edge Computing, Key concepts and characteristics, Motivations and Advantages: Reasons for adopting Fog and Edge Computing, Comparative advantages over traditional Cloud Computing, Use Cases and Applications: Challenges and Limitations: Security concerns, privacy issues, and compliance, Comparison with Cloud Computing: Contrasting characteristics and use cases, Complementary roles in a hybrid computing environment.

UNIT II ARCHITECTURES AND COMPONENTS   **6**

Edge Computing Architecture Models: Design principles and characteristics, Hierarchical vs. flat architectures. Fog Nodes and Infrastructure: Role and capabilities of Fog nodes, Infrastructure components supporting Edge Computing. Edge Device Types and Capabilities: Categorization of Edge devices (e.g., gateways, sensors, actuators), Capabilities and limitations of different device types. Edge-to-Cloud Communication Models; Middleware and Software Frameworks: Middleware solutions for Edge Computing.

UNIT III SECURITY AND PRIVACY IN EDGE COMPUTING   **6**

Security Challenges in Edge Computing: Identifying security threats in Edge Computing, Risks associated with decentralized architectures. Methods for secure access control: Role of authentication in ensuring device integrity, Data Encryption in Edge Environments: Encryption strategies for data at rest and in transit. Privacy Concerns and Regulatory Compliance; Risk Management and Incident Response.

UNIT IV EDGE APPLICATION DEVELOPMENT   **6**

Programming Models and Frameworks: Overview of programming models for Edge Computing, Exploration of popular frameworks for Edge application development, Integration with IoT Devices:
Techniques for integrating Edge Computing with Internet of Things (IoT) devices. Edge Application Deployment Strategies; Edge-to-Cloud Communication Patterns; Optimizing Edge Application Performance: Performance considerations in Edge Computing, Strategies for optimizing resource usage and responsiveness.

UNIT V PERFORMANCE OPTIMIZATION AND FUTURE TRENDS   **6**

Performance Metrics and Optimization: Metrics for evaluating performance in Edge Computing
Strategies for optimizing Edge applications; Energy-Efficient Edge Architectures: Techniques for optimizing energy consumption, Green computing practices in Edge environments, Real-time Processing and Analytics: Approaches for real-time data processing at the Edge, Analytics capabilities in decentralized architectures. Scalability and Flexibility in Edge Architectures; Emerging Trends in Fog and Edge Computing.

TOTAL NUMBER OF PERIODS INCLUDING LAB: 60

SAMPLE LIST OF EXPERIMENTS   NUMBER OF PRACTICAL PERIODS: 30

1. Design and deploy a fog computing architecture for a smart city application.
2. Optimize data processing algorithms for edge devices in a real-time monitoring system.
3. Evaluate resource utilization in a fog computing environment using simulation tools.
4. Implement security protocols for edge devices to protect sensitive data.
5. Develop a scalable edge computing solution for IoT devices in a manufacturing setting.
6. Integrate fog computing with cloud services to enhance overall system efficiency.
7. Perform a hands-on deployment of edge computing nodes in a network infrastructure.
8. Design and implement a fault-tolerant edge computing solution for critical applications.
9. Conduct performance testing to assess the latency reduction achieved through edge computing.
10. Create a comprehensive case study on the successful integration of fog and edge computing in a specific industry.

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Proficiently deploy and manage fog and edge computing solutions in diverse environments.
CO2   Demonstrate expertise in optimizing data processing and analytics at the network edge.
CO3   Evaluate and enhance resource efficiency for fog computing deployments.
CO4   Implement robust security measures for safeguarding edge computing data.
CO5   Successfully integrate fog and edge computing technologies to address real-world challenges.

TEXT& REFERENCE BOOKS

1. "Fog and Edge Computing: Principles and Paradigms" by Rajkumar Buyya and Amir Vahid Dastjerdi (Wiley, 1st Edition, 2018),
2. "Edge Computing: A Primer" by Shiwen Mao and Ying-Chang Liang (Wiley, 1st Edition, 2019),
3. "Fog Computing in the Internet of Things: Intelligence at the Edge" by Yogesh Simmhan, Nalini Venkatasubramanian, and Viktor K. Prasanna (Springer, 1st Edition, 2018),
4. "Fog Computing: Concepts, Frameworks and Technologies" by Flavio Bonomi, Rodolfo Milito, Jiang Zhu, and Sateesh Addepalli (Springer, 1st Edition, 2018),
5. "Edge Computing – The Dawn of Industry 4.0 Revolution" by Satya Prakash Ghrera (CRC Press, 1st Edition, 2020)
6. "Fog Computing: Towards Minimizing Latency and Improving QoE in the Internet of Things" by Mohammad Mehedi Hassan, Saad B. Qaisar, and Abdelgadir Mohammed Ahmed (Springer, 1st Edition, 2018).

Course Code	BLOCKCHAIN TECHNOLOGY AND CLOUD COMPUTING	L	T	P	C
CSPE3XV58		2	0	2	3

COURSE OBJECTIVES:

1. Introduce students to the core principles, mechanisms, and applications of Blockchain Technology and CloudComputing.

2. Foster a solid theoretical foundation in Blockchain mechanisms, cryptographic principles, cloud architectures,and enabling technologies.

3. Provide hands-on experience through lab sessions, experiments, and projects to apply theoretical conceptsinto practical scenarios.

4. Enable students to integrate the principles and practices of Blockchain and Cloud Computing to developinnovative and efficient solutions.

5. Empower students to analyse real-world use cases and explore the potential of utilizing Blockchain and CloudComputing in various applications.

UNIT I INTRODUCTION TO BLOCKCHAIN AND CLOUD COMPUTING　　　　**6**

Blockchain Technology Mechanisms & Networks: Overview, Blockchain Networks, Decentralization. **Transactions, Blocks, P2P Systems:** Structure of Transactions, Concept of Blocks, Peer-to-Peer (P2P) Systems. **Keys, Digital Signatures, Hashing:** Cryptographic Keys, Digital Signatures in Blockchain Transactions, Hash Functions and Applications. **Cloud Computing - Definition, Evolution, Characteristics**: Introduction to Cloud Computing, Characteristics, Evolution of Cloud. **Principles of Parallel Computing, Elasticity:** Principles of Parallel Computing in Cloud, Elasticity and Scalability in Cloud Environments, Techniques for Parallelism. **On-demand Provisioning:** Understanding On-demand Provisioning in Cloud, Resource Allocation and De-allocation, Implementing Scalable Resources on Demand.

UNIT II CYPTOCURRENCY AND CLOUD ENABLING TECHNOLOGY　　　　**6**

Bitcoin Overview: Introduction to Bitcoin, Bitcoin Mining Process, Bitcoin Wallets: Types and Functions, Ethereum Virtual Machine (EVM). **Blockchain & Cryptocurrency:** Consensus Mechanisms in Blockchain, Smart Contracts and Applications, Decentralization and its Significance, Impact of Blockchain Technology on Cryptocurrency. **Understanding Cloud Basics:** Service-Oriented Architecture (SOA), Fundamentals of Virtualization, Types and Levels of Virtualization, Cloud Characteristics and Attributes, Elasticity and Scalability in Cloud Computing. **Principles and Implementation**: Cloud Services and Deployment Models, Cloud Virtualization Tools and Mechanisms. CPU, Memory, and I/O Devices Virtualization, Disaster Recovery in Cloud Environments.

UNIT III ETHEREUM & CLOUD ARCHITECTURE　　　　**6**

Overview of Ethereum: Ethereum Origins and Core Principles, Smart Contracts, Decentralization. **Ethereum Operations & Transactions:** Ethereum Accounts: Types and Functions, Transaction Process in Ethereum Network, Ethereum Smart Contracts: Structure and Execution. **Understanding Hyperledger:** Introduction to Hyperledger Fabric, Hyperledger Composer: Overview and Functions, Consensus Mechanisms in Hyperledger. **Layered Cloud ArchitectureDesign:** Components of Cloud Architecture, Exploring Layers in Cloud Environments, Architectural Design Challenges. **Cloud Services & Service Models:** Overview of Cloud Services (IaaS, PaaS, SaaS), Characteristics and Applications of Each Service Model, Challenges and Solutions in Service Deployment. **Cloud Storage Solutions:** Cloud Storage Basics and Principles, Storage-as-a-Service (STaaS) Concept, Analysis of Cloud Storage Providers (e.g., S3).

UNIT IV SOLIDITY PROGRAMMING AND CLOUD SECURITY **6**

Solidity Programming Language: Syntax, Types, Variables, Functions, Data Structures in Solidity. **Ethereum Wallet & Smart Contracts:** Overview of Ethereum Wallet, Structure & Working of Smart Contracts. **Basics of Solidity & Smart Contract Structure:** Understanding Basic Solidity Concepts, Components and Structure of Smart Contracts. **Resource Provisioning & Cloud Overview:** Cloud Resource Allocation & Management, Cloud Service Models (IaaS, PaaS, SaaS). **Security Overview in Cloud Environment:** Cloud Security Challenges, Security Governance in Cloud. **IAM, Virtual Machine Security, Security Standards:** Identity & Access Management in Cloud, Ensuring Security in Virtual Machines, Compliance with Security Standards in Cloud Environments.

UNIT V BLOCKCHAIN APPLICATIONS AND ADVANCEMENT OF CLOUD TECHNOLGIES 6

IoT Integration with Blockchain: Introduction to IoT (Internet of Things), Blockchain's Role in IoT Security, IoT Data Management using Blockchain. **Blockchain in Medical Record Management:** Healthcare Data Challenges & Security, Blockchain's Impact on Medical Records, Privacy, Integrity, and Accessibility in Medical Records, Applications. **Domain Name Service (DNS) using Blockchain:** DNS Issues in Centralized Systems, Blockchain-Based Decentralized DNS. **Alt Coins and Their Significance:** Introduction to Alt Coins (Alternative Cryptocurrencies), Role and Market Impact of Alt Coins, Differentiating Factors and Advantages, Future Trends and Adoption of Alt Coins. **Future Trends in Blockchain Technology:** Emerging Trends, Impact on Industries and Potential Evolution. **Hadoop and Big Data Processing in Cloud:** Introduction to Hadoop and MapReduce, Cloud-Based Big Data Analytics, Hadoop's Role in Cloud Computing, Use Cases and Industry Applications. **Google App Engine (GAE) for Cloud Development:** Overview of GAE, Building and Deploying Applications on GAE, Scalability and Maintenance Aspects, GAE's Contribution to Cloud Development. **OpenStack and Cloud Federation:** OpenStack in Cloud Environments, Federation in Cloud Computing, Multi-Cloud Management using OpenStack, Benefits and Challenges of Cloud Federation.

TOTAL NUMBER OF PERIODS INCLUDING LAB: 60

SAMPLE LIST OF EXPERIMENTS   NUMBER OF PRACTICAL PERIODS: 30

1. Simulate a basic blockchain network to understand transactions, blocks, and the structure of peer-to-peer systems, alongside implementing on-demand resource provisioning in cloud computing.

2. Analyze the Bitcoin mining process, types of wallets, and Ethereum Virtual Machine (EVM) while exploring consensus mechanisms and smart contracts' impact on cryptocurrency, and understanding cloud basics such as service-oriented architecture (SOA) and virtualization.

3. Develop smart contracts in Solidity programming language for Ethereum, comprehend Hyperledger Fabric and Hyperledger Composer, and design layered cloud architecture considering cloud services and service models.

4. Implement Solidity programming language features like variables, functions, and data structures, alongside understanding Ethereum wallet and smart contracts structure, and ensuring cloud security through identity & access management (IAM) and compliance with security standards.

5. Explore blockchain applications like IoT integration and medical record management, analyze alternative cryptocurrencies (Alt Coins) and their significance, and investigate

advancements in cloud technologies such as Hadoop for big data processing and Google App Engine (GAE) for cloud development.

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Demonstrate a comprehensive understanding of the fundamental principles and mechanisms of BlockchainTechnology and Cloud Computing.

CO2   Apply theoretical knowledge to solve problems related to Blockchain and Cloud Computing through hands-onexperiments and practical sessions.

CO3   Integrate Blockchain Technology and Cloud Computing for the development of innovative solutions,leveraging the combined strengths of both domains.

CO4   Exhibit problem-solving skills by analysing and proposing solutions using Blockchain and Cloud Computingprinciples in diverse scenarios.

CO5   Analyse and evaluate real-world use cases to determine the suitability and potential impact of Blockchain andCloud Computing technologies.

CO6   Develop practical proficiency in executing projects that utilize Blockchain and Cloud Computing for addressingcontemporary challenges.

TEXTBOOKS:
1. Daniel Drescher, "Blockchain Basics: A Non-Technical Introduction in 25 Steps", 1st Edition, 2017
2. Andreas M. Antonopoulos, "Mastering Bitcoin: Unlocking Digital Cryptocurrencies", 2nd Edition, 2017
3. Stephen Grider, "Ethereum Programming", 1st Edition, 2020.
4. Thomas Erl, Ricardo Puttini, and Zaigham Mahmood, "Cloud Computing: Concepts, Technology &Architecture", 2nd edition, 2013.
5. Michael J. Kavis, "Architecting the Cloud: Design Decisions for Cloud Computing Service Models", 1st edition,2014.

REFERENCES:

1. Don Tapscott and Alex Tapscott, "Blockchain Revolution: How the Technology Behind Bitcoin and OtherCryptocurrencies is Changing the World", 2nd edition, 2018.
2. Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller, and Steven Goldfeder, "Bitcoin andCryptocurrency Technologies: A Comprehensive Introduction", 1st edition, 2016.
3. Judith Hurwitz, Robin Bloor, Marcia Kaufman, and Fern Halper, "Cloud Computing for Dummies", 1st edition,2009.
4. Tim Mather, Subra Kumaraswamy, and Shahed Latif, "Cloud Security and Privacy: An Enterprise Perspectiveon Risks and Compliance", 1st edition, 2009.

IOT SYSTEMS

Course Code	FOUNDATIONS OF EMBEDDED IOT SYSTEMS	L	T	P	C
CSPE3XV61		2	0	2	3

COURSE OBJECTIVES:

1. To provide students with a good depth of knowledge of Designing Embedded and IOT Systems for various applications.
2. Knowledge of the design and analysis of Embedded and IoT Systems for Electronics Engineering students.

UNIT I INTRODUCTION TO EMBEDDED AND IOT SYSTEMS   **6**

Introduction Embedded and IoT systems, Definition, Examples and components of embedded and IoT Systems, Embedded and IoT Systems Design Process, Various Embedded and IoT cores controllers.

UNIT II HARDWARE/SOFTWARE CO-DESIGN FOR EMBEDDED AND IOT SYSTEMS   **6**

Microcontrollers for embedded systems, Arduino embedded platform, Peripheral interfacing and programming with Arduino platform, Sensors and Actuator interfacing, Cloud support with Arduino platform.

UNIT III PROTOCOLS FOR EMBEDDED AND IOT SYSTEMS   **6**

Serial protocols, UART, I2C, and SPI. NFC, Wireless protocols like, RFID, Zig-bee, IEEE 802.15.4e, Thread, 6LoWPAN, Constrained Application Protocol (CoAP), Extensible Messaging Protocol (XMPP) , WebSocket , Advanced Message Queueing Protocol (AMQP) , Message Queue Telemetry Transport (MQTT), Web Real Time Communications (WebRTC), LoRa, SIGFOX, Z Wave.

UNIT IV IOT BASED EMBEDDED SYSTEMS   **6**

Open source OS for IoT such as Contiki OS, TinyOS, Basic architecture of an IoT based Embedded Systems., Embedded Hardware for IoT applications, like Raspberry Pi, Arduino, and Raspberry Pi based development board, IoT Cloud Platform and IoT client applications on mobile phones.

UNIT V CASE STUDIES OF EMBEDDED AND IOT SYSTEMS   **6**

Embedded application development through Arduino and Raspberry Pi based development boards, Development of mini-Project on new version of Operating systems and development board. That project should also address to the current societal needs.

TOTAL NUMBER OF PERIODS INCLUDING LAB: 60

SAMPLE LIST OF EXPERIMENTS   **NUMBER OF PRACTICAL PERIODS: 30**

1. Study of Open source operating system used in Embedded Design.

2. Introduction to Arduino based Embedded System Programming.

3. LED Interfacing program for Arduino based Embedded System

4. Interfacing Push button Switch interfacing with Arduino based Embedded System

5. External Peripheral Interfacing with Arduino based Embedded System.

6. On Chip peripheral programming with Arduino/Raspberry Pi based Embedded System

7. Serial Communication Protocol programming with Arduino/Raspberry Pi based Embedded Systems.

8. Wireless communications with Arduino/Raspberry Pi Embedded IOT Platform.

9. Bluetooth communication interfacing with Arduino/Raspberry PiEmbedded IOT Board.

10. WiFi module interfacing with Arduino/Raspberry Pi Embedded IOT Board.

11. Embedded Systems design with IOT capability.

12. IOT based Temperature monitoring embedded system with open source cloud tools.

13. Introduction to RTOS

14. RTOS based task performances

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Knowledge of theory and practice related to Embedded and IOT System.

CO2   Ability to identify, formulate and solve engineering problems by using Embedded Systems with IoT.

CO3   Ability to implement real field problem by gained knowledge of Embedded Systems with IoT capability.

TEXT CUM REFERENCE BOOKS:

1. Muhammad Ali Mazidi Shujen Chen, Sepehr Naimi Sarmad Naimi "Embedded Programming
2. Using C Language", 1st Edition, Freescale ARM Cortex-M.
3. Steve Ferbur, "ARM System on Chip".
4. Rajkamal, "Embedded System: Architecture, Programming and Design", TMH3.
5. Dr. OvidiuVermesan, Dr. Peter Friess, "Internet of Things: Converging Technologies for
6. Smart Environments and Integrated Ecosystems", River Publisher

Course Code	IOT NETWORKS	L	T	P	C
CSPE3XV62		2	0	2	3

COURSE OBJECTIVES:

1. To give an understanding about the choice and application of IoT & M2M communication protocols.
2. To describe Cloud computing and design principles of IoT.
3. To relate to MQTT clients, MQTT server and its programming.
4. To describe the architectures and communication protocols of WSNs.
5. To identify the uplink and downlink communication protocols associated with specific application of IOT /WSNs.

UNIT I OVERVIEW OF INTERNET OF THINGS   **6**

IoT Conceptual Framework, IoT Architectural View, Technology Behind IoT, Sources of IoT,M2M communication, Examples of IoT. Modified OSI Model for the IoT/M2M Systems, data enrichment, data consolidation and device management at IoT/M2M Gateway, web communication protocols used by connected IoT/M2M devices, Message communication protocols (CoAP-SMS, CoAP-MQ, MQTT,XMPP) for IoT/M2M devices. -Refer Chapter 1,2 and 3 of Text 1.

UNIT II ARCHITECTURE AND DESIGN PRINCIPLES FOR IOT   **6**

Internet connectivity, Internet based communication, IPv4, IPv6,6LoWPAN protocol, IP Addressing in the IoT, Application layer protocols: HTTP, HTTPS,FTP, TELNET and ports.

UNIT III PROTOTYPING AND DESIGNING SOFTWARE FOR IOT APPLICATIONS   **6**

Introduction, Prototyping Embedded device software, Programming Embedded Device Arduino Platform using IDE, Reading data from sensors and devices, Devices, Gateways, Internet and Web/Cloud services software development. Programming MQTT clients and MQTT server. Introduction to IoT privacy and security. Vulnerabilities, security requirements and threat analysis, IoT Security Tomography and layered attacker model. -Refer Chapter 9 and 10 of Text 1.

UNIT IV OVERVIEW OF WIRELESS SENSOR NETWORKS   **6**

Challenges for Wireless Sensor Networks, Enabling Technologies for Wireless Sensor Networks. Architectures: Single-Node Architecture - Hardware Components, Energy Consumption of Sensor Nodes, Operating Systems and Execution Environments, Network Architecture-Sensor Network Scenarios, Optimization Goals and Figures of Merit, Design principles for WSNs, Service interfaces of WSNs Gateway Concepts. -Refer Chapter 1,2, 3 of Text 2.

UNIT V COMMUNICATION PROTOCOLS   **6**

Physical Layer and Transceiver Design Considerations, MAC Protocols for Wrreless Sensor Networks, Low Duty Cycle Protocols and Wakeup Concepts - S-MAC, The Mediation Device Protocol, Wakeup Radio Concepts, Contention based protocols (CSMA, PAMAS), Schedule based protocols (LEACH, SMACS, TRAMA) Address and Name Management in WSNs, Assignment of MAC Addresses, Routing Protocols- Energy-Efficient Routing, Geographic Routing, Hierarchical networks by clustering. Refer Chapter 4, 5, 7 and 11 of Text 2. Lt, L2, L3

TOTAL NUMBER OF PERIODS INCLUDING LAB: 60

SAMPLE LIST OF EXPERIMENTS   NUMBER OF PRACTICAL PERIODS: 30

1. Implement an IoT conceptual framework and analyze M2M communication using CoAP-SMS and CoAP-MQ protocols, focusing on data enrichment and consolidation at the IoT/M2M gateway.

2. Design a prototype for Internet connectivity in IoT, exploring IPv4, IPv6, and 6LoWPAN protocol, alongside understanding application layer protocols like HTTP, HTTPS, and FTP.

3. Develop embedded device software using Arduino IDE, read data from sensors, and program MQTT clients and servers while addressing IoT privacy and security concerns through threat analysis and IoT Security Tomography.

4. Explore enabling technologies and architectures for wireless sensor networks (WSNs), understand single-node architecture components, and optimize energy consumption using operating systems and execution environments.

5. Analyze physical layer and transceiver design considerations for WSNs, implement MAC protocols like S-MAC and contention-based protocols like CSMA, and design energy-efficient routing protocols for hierarchical networks by clustering

COURSE OUTCOMES:

On completion of this course, the students will be able to:
.
CO1   Understand choice and application of IoT & M2M communication protocols.
CO2   Describe Cloud computing and design principles of IoT.
CO3   Relate to MQTT clients, MQTT server and its programming.
CO4   Describe the architectures and communication protocols of WSNs.
CO5   Identify the uplink and downlink communication protocols associated with specific application of IOT /WSNs.

TEXTBOOKS:

1. Raj Kamal, "Internet of Things-Architecture and Design Principles", McGraw Hill Education.
2. Holger Karl & Andreas Willig, " Protocols And Architectures for Wireless Sensor Networks", John Wiley, 2005.

REFERENCES:

1. Feng Zhao & Leonidas J. Guibas, "Wireless Sensor Networks- An Information Processing Approach", Elsevier, 007.
2. Kazem Sohraby, Daniel Minoli, & Taieb Znati, " Wireless Sensor Networks- Technology, Protocols and Applications", John Wiley, 2007.
3. 3. Anna Hac, 'Wireless Sensor Network Designs", John Wiley, 2003.

Course Code	SECURE HARDWARE AND EMBEDDED DEVICES	L	T	P	C
CSPE3XV63		2	0	2	3

COURSE OBJECTIVES:

1. To Understand the Concepts of Computer and Network Security
2. To Study and Understand Encryption Techniques.
3. To Explore the different aspects of Embedded System Security.
4. To Understand the role of Security Aspects during Data Transfer and Communication.
5. To apply the Security Algorithms for Real-time Applications.

UNIT I BACKGROUND AND INTRODUCTION   **6**

Computer and Network Security Concepts: Computer Security Concepts - The OSI Security Architecture - Security Attacks - Security Services - Security Mechanisms - Fundamentals of Security Design Principles - Attack Surfaces and Attack Trees - A Model for Network Security. Introduction to Number Theory: Divisibility and the Division Algorithm - The Euclidean Algorithm - Modular Arithmetic - Prime Numbers - Fermat's and Euler's Theorems - Testing for Primality - TheChinese Remainder Theorem - Discrete Logarithms.

UNIT II SYMMETRIC CIPHERS   **6**

Classical Encryption Techniques: Symmetric Cipher Model - Substitution Techniques - Transposition Techniques. Block Ciphers and the Data Encryption Standard (DES): Traditional Block Cipher Structure - The Data Encryption Standard - A DES Example - Strength of DES. Advanced Encryption Standard: Finite Field Arithmetic - AES Structure - AES Transformation Functions - AES Key Expansion - An AES Example - AES Implementation.

UNIT III EMBEDDED SYSTEMS SECURITY   **6**

Embedded Security Trends - Security Policies - Security Threats. System Software Considerations: The Role of Operating System - Microkernel versus Monolithic - Core Embedded OS Security Requirements - Access Control and Capabilities - Hypervisors and System Virtualization - I/O Virtualization - Remote Management - Assuring Integrity of the TCB.

UNIT IV EMBEDDED CRYPTOGRAPHY AND DATA PROTECTION PROTOCOLS   **6**

The One-time Pad - Cryptographic Modes - Block Ciphers - Authenticated Encryption - Public KeyCryptography - Key Agreement - Public Key Authentication - Elliptic Curve Cryptography - Cryptographic Hashes - Message Authentication Codes - Random Number Generation - Key Management for Embedded Systems - Cryptographic Certifications. Data Protection Protocols for Embedded Systems: Data-in-Motion Protocols - Data-at-Rest Protocols. Emerging Applications: Embedded Network Transactions - Automotive Security - Secured Android.

UNIT V PRACTICAL EMBEDDED SYSTEM SECURITY   **6**

Network Communications Protocols and Built-in Security - Security Protocols and Algorithms - The Secured Socket Layer - Embedded Security - Wireless - Application-Layer and Client/Server Protocols - Choosing and Optimizing Cryptographic Algorithms for Resource-Constrained Systems - Hardware Based Security.

TOTAL NUMBER OF PERIODS INCLUDING LAB: 60

SAMPLE LIST OF EXPERIMENTS **NUMBER OF PRACTICAL PERIODS:** **30**

1. Write a program, to Analyses OSI Security Architecture and its role in computer security.
2. Write a program to Implement classical encryption techniques such as substitution and transposition.
3. Write a program to implement the structure and strength of the Data Encryption Standard (DES).
4. Write a program to Explore the Advanced Encryption Standard (AES) and implement AES encryption.
5. Write a program to Investigate embedded security trends, policies, and threats.
6. Examine system software considerations for embedded systems, including access control and hypervisors.
7. Write a program to Implement cryptographic primitives such as the one-time pad and cryptographic modes.
8. Write a program to implement public key cryptography and its applications in embedded systems.
9. Write a program to Compare and contrast data protection protocols for embedded systems.
10. Write a program to implement a secure communication protocol for resource-constrained embedded systems.

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Understand the significance of Security.
CO2   Understand the major concepts and techniques related to Cryptography.
CO3   Demonstrate thorough knowledge about the aspects of Embedded System Security.
CO4   Delivers insight onto role of Security Aspects during Data Transfer and Communication.
CO5   Applying the Security Algorithms for Real-time Applications.

TEXT CUM REFERENCE BOOKS:

1. "Cryptography and Network Security Principles and Practice", 7th Edition - Global Edition, William Stallings, Pearson Education Limited, 2017.
2. "Embedded Systems Security - Practical Methods for Safe and Secure Software and Systems Development", David Kleidermacher and Mike Kleidermacher, Newnes (an imprint of Elsevier), 2012.
3. "Practical Embedded Security - Building Secure Resource-Constrained Systems", Timothy Stapko, Newnes (an imprint of Elsevier), 2008.

Course Code	**IOT PROCESSORS**	L	T	P	C
CSPE3XV64		2	0	2	3

COURSE OBJECTIVES:

1. Learn the architecture and features of ARM.
2. Study the exception handling and interrupts in CORTEX M3
3. Program the CORTEX M3
4. Learn the architecture of STM 32L15XXX ARM CORTEX M3/M4 microcontroller.
5. Understand the concepts of System – On – Chip (SoC)

UNIT I OVERVIEW OF ARM AND CORTEX-M3   **6**

ARM Architecture – Versions, Instruction Set Development, Thumb 2 and Instruction Set Architecture, Cortex M3 Basics: Registers, Stack Pointer, Link Register, Program Counter, Special Registers, Operation Mode, Exceptions and Interrupts, Vector Tables, Stack Memory Operations, Reset Sequence , CORTEX M3 Instruction Sets: Assembly Basics, Instruction List, Instruction Descriptions, CORTEX M3 – Implementation Overview: Pipeline, Block Diagram. Bus Interfaces, I –Code Bus, D – Code Bus, System Bus- External PPB and DAP Bus.

UNIT II CORTEX EXCEPTION HANDLING AND INTERRUPTS   **6**

Exception Types, Priority, Vector Tables, Interrupt Inputs and Pending behaviour, Fault Exceptions, Supervisor Call and Pendable Service Call, NVIC: Nested Vector Interrupt Controller, Overview, Basic Interrupts, SYSTICK Time, Interrupt Behaviourm Interrupt/Exception Sequences, Exception Exits, Nested Interrupts, Tail – Chaining Interrupts, Late Arrivals and Interrupt Latency.

UNIT III CORTEX M3/M4 PROGRAMMING   **6**

Cortex M3/M4 Programming: Overview, Typical Development Flow, Using C, CMSIS UsingAssembly, Excepiton Programming Using Interrupts, Exception/Interrupt Handlers, Software Interrupts, Vector Table Relocation, Memory Protection Unit and other CORTEX M3 Features, MPU Registers, Setting up the MPU, Power Management, Multiprocessor Configuration.

UNIT IV STM32L15XXX ARMCORTEX M3/M4 MICROCONTROLLER AND DEBUGGING TOOLS   **6**

STM32L15XXX ARM CORTEX M3/M4 Microcontroller: Memory and Bus Architecture, Power Control, Reset and Clock Control, STM32L15XXX Peripherals: GPIOs, System Configuration Controller, NVIC, ADC, Comparators, GP Timers, USART Development and Debugging Tools: Software and Hardware tools like Cross Assemblerm Compiler, Debugger, Simulator, In – Circuit Emulator (ICE), Logic Analyser.

UNIT V INTRODUCTION TO SYSTEM – ON – CHIP   **6**

System Architecture: An Overview, Components of the System Processors, Memories and Interconnects, Processor Architectures, Memory and Addressing, System Level Interconnection –An Approach for SOC Design – Chip basics – Cycle Time – Die Area – Power and Cost – Area, Power and Time Trade – Offs in Processor Design – Reliability and Configurability – SOC Design Approach – Application Studies – AES, 3D Graphics Processor. Image Compression and Video Compression.

TOTAL NUMBER OF PERIODS INCLUDING LAB: 60

SAMPLE LIST OF EXPERIMENTS   NUMBER OF PRACTICAL PERIODS: 30

ARM Assembly Programming:

1. Write a program to add two 32-bit numbers stored in r0 and r1 registers and write the result to r2. The result is stored to a memory location. a) Run the program with breakpoint and verify the result b) Run the program with stepping and verify the content of registers at each stage.

2. Write ARM assembly to perform the function of division. Registers r1 and r2 contain the dividend and divisor, r3 contains the quotient, and r5 contains the remainder.

3. Embedded C Programming on ARM Cortex M3/M4 Microcontroller

4. Write a program to turn on green LED (Port B.6) and Blue LED (Port B.7) on STM32L-Discovery by configuring GPIO.

5. Transmit a string "Programming with ARM Cortex" to PC by configuring the registers of USART2. Use polling method.

6. ARM Cortex M3/M4 Programming with CMSIS

7. Write a program to toggle the LEDs at the rate of 1 sec using standard peripheral library. Use Timer3 for Delay.

8. Transmit a string "Programming with ARM Cortex" to PC by using standard peripheral library with the help of USART3. Use polling method.

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Explain the architecture and features of ARM.

CO2   List the concepts of exception handling.

CO3   Write a program using ARM CORTEX M3/M4.

CO4   Learn the architecture of STM32L15XXX ARM CORTEX M3/M4.

CO5   Design an SoC for any application.

TEXT CUM REFERENCE BOOKS

1. Joseph Yiu, The Definitive Guide to the ARM CORTEX M3/M4, Second Edition, Elsevier, 2010.(Unit – I, II)
2. Andrew N Sloss, Dominic Symes, Chris Wright, ARM System Developers Guide Designing and Optimising System Software, Elsevier, 2006 (Unit – III, IV)
3. 3. Michael J Flynn and Wayne Luk, Computer System Design, System On Chip, Wiley India 2011.(Unit – V)

Course Code	MOBILE APPLICATION DEVELOPMENT	L	T	P	C
CSPE3XV65		2	0	2	3

COURSE OBJECTIVES:

1. To facilitate students to understand android SDK
2. To help students to gain basic understanding of Android application development
3. To understand how to work with various mobile application development frameworks
4. To inculcate working knowledge of Android Studio development tool
5. To learn the basic and important design concepts and issues of development of mobile applications.

UNIT I MOBILE PLATFORM AND APPLICATIONS   **6**

Mobile Device Operating Systems — Special Constraints & Requirements — Commercial Mobile Operating Systems — Software Development Kit: iOS, Android, BlackBerry, Windows Phone —MCommerce — Structure — Pros & Cons — Mobile Payment System — Security Issues

UNIT II INTRODUCTION TO ANDROID   **6**

Introduction to Android: The Android Platform, Android SDK, Eclipse Installation, Android Installation, Building you First Android application, Understanding Anatomy of Android Application, Android Manifest file.

UNIT III ANDROID APPLICATION DESIGN ESSENTIALS   **6**

Anatomy of Android applications, Android terminologies, Application Context, Activities, Services, Intents, Receiving and Broadcasting Intents, Android Manifest File and its common settings, Using Intent Filter, Permissions.

UNIT IV ANDROID USER INTERFACE DESIGN & MULTIMEDIA   **6**

User Interface Screen elements, Designing User Interfaces with Layouts, Drawing and Working with Animation. Playing Audio and Video, Recording Audio and Video, Using the Camera to Take and Process Pictures

UNIT V ANDROID APIs   **6**

Using Android Data and Storage APIs, Managing data using Sqlite, Sharing Data between Applications with Content Providers, Using Android Networking APIs, Using Android Web APIs, Using Android Telephony APIs, Deploying Android Application to the World.

    TOTAL NUMBER OF PERIODS INCLUDING LAB: 60

SAMPLE LIST OF EXPERIMENTS   NUMBER OF PRACTICAL PERIODS: 30

1. Develop an application that uses GUI components, Font, Layout Managers and

event listeners.

2. Develop an application that makes use of databases

3. Develop a native application that uses GPS location information

4. Implement an application that creates an alert upon receiving a message

5. Develop an application that makes use of RSS Feed.

6. Create an application using Sensor Manager

7. Create an android application that converts the user input text to voice.

8. Develop a Mobile application for simple and day to day needs (Mini Project)

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1	Identify various concepts of mobile programming that make it unique from programming for other platforms.
CO2	Create, test and debug Android application by setting up Android development.
CO3	Demonstrate methods in storing, sharing and retrieving data in Android applications
CO4	Utilize rapid prototyping techniques to design and develop sophisticated mobile interfaces
CO5	Create interactive applications in android using databases with multiple activities including audio, video and notifications and deploy them in marketplace

TEXT CUM REFERENCE BOOKS:

1. Lauren Darcey and Shane Conder, "Android Wireless Application Development", Pearson Education, 2nd ed. (2011)

2. Google Developer Training, "Android Developer Fundamentals Course – Concept Reference", Google Developer Training Team, 2017.

3. Prasanth Kumar Pattnaik,Rajib Mall,"Fundamentals of Mobile Computing",PHI Learning Pvt.Ltd,New Delhi-2012

4. Reto Meier, "Professional Android 2 Application Development", Wiley India Pvt Ltd, 2010

5. Mark L Murphy, "Beginning Android", Wiley India Pvt Ltd, 2009

6. Dawn Griffiths and David Griffiths, "Head First Android Development", 1st Edition, O"Reilly SPD Publishers, 2015. ISBN-13: 978-9352131341

7. Erik Hellman, "Android Programming – Pushing the Limits", 1st Edition, Wiley India Pvt Ltd, 2014. ISBN-13: 978-8126547197.

8. Bill Phillips, Chris Stewart and Kristin Marsicano, "Android Programming: The Big Nerd

9. Ranch Guide", 4th Edition, Big Nerd Ranch Guides, 2019. ISBN-13: 978-0134706054

Course Code	INDUSTRIAL IOT & HEALTHCARE SYSTEMS	L	T	P	C
CSPE3XV66		2	0	0	2

COURSE OBJECTIVES:

1. To teach key skills employed in the IIoT & IoRT space building applications.
2. To give knowledge on Design suitable network architecture and use appropriate learning algorithm.
3. To Comprehend IOT protocols
4. To implement digital Twin
5. To implement IOT systems for robotics

UNIT I INTRODUCTION TO INDUSTRIAL IOT   **6**

Technical requirements, IoT Background-History and definition, IoT enabling factors, IoT applications, IoT key technologies, I-IoT, IoT and I-IoT – similarities and differences, Industry environments and scenarios covered by I-IoT.

UNIT II UNDERSTANDING THE INDUSTRIAL PROCESS AND DEVICES TECHNICAL   **6**
REQUIREMENTS

The industrial process-Automation in the industrial process, Control and measurement systems, Types of industrial processes.

UNIT III INDUSTRIAL DATA FLOW AND DEVICES   **6**

Technical requirements, The I-IoT data flow in the factory, Measurements and the actuator chain Sensors, The converters - Digital to analogical, Analog to digital, Actuators, Controllers - Microcontrollers, Embedded microcontrollers, Microcontrollers with external memory, DSP's. Industrial protocols -Automation networks, the fieldbus, Developing Industrial IoT and Architecture Introduction to the I-IoT platform and architectures, OSGi, micro service, containers, and server less computing, The standard IoT flow.

UNIT IV INTRODUCTION TO IOT BASED HEALTH CARE   **6**

Introduction to IoT applications in smart healthcare& their distinctive advantages - Patient Health Monitoring System (PHMS), Tele-Health, Tele-medicine, Tele-Monitoring, Mobile Health Things (m-health).

UNIT V IOT SMART SENSING HEATH CARE AND POWER CHALLENGE   **6**

Concept of Generic Biomedical sensors, Smart Sensors: Monitor health parameters, Wearable ECG sensors, IoT Data Acquisition System, Energy harvesting, Battery based systems, Power management.

TOTAL NUMBER OF PERIODS INCLUDING LAB: **60**

SAMPLE LIST OF EXPERIMENTS   **NUMBER OF PRACTICAL PERIODS:**   **30**

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Understand key skills employed in the IIoT & IoRT space building applications.
CO2   Design suitable network architecture and use appropriate learning algorithm.
CO3   Comprehend IOT protocols

CO4   Implement digital Twin
CO5   Implement IOT systems for robotics

1. Analyze the technical requirements and key technologies of Industrial IoT (I-IoT), comparing its similarities and differences with traditional IoT, while exploring its applications and enabling factors.

2. Explore automation, control, and measurement systems in industrial processes, categorizing different types of industrial processes.

3. Investigate the industrial data flow in factories, focusing on sensors, converters, actuators, controllers, and industrial protocols, and introduce I-IoT platform architectures like OSGi and microservices.

4. Examine IoT applications in healthcare, including Patient Health Monitoring System (PHMS), Tele-Health, Tele-medicine, and Mobile Health Things (m-health), highlighting their advantages.

5. Implement smart sensing healthcare solutions using generic biomedical sensors, wearable ECG sensors, and IoT Data Acquisition Systems, while addressing power challenges through energy harvesting, battery-based systems, and power management techniques.

TEXT CUM REFERENCE BOOKS:

1. "Industry 4.0: The Industrial Internet of Things", Alasdair Gilchrist, Apress,2016

2. "Introduction to Industrial Internet of Things and Industry 4.0", Sudip Misra, Chandana Roy, Anadarup Mukherjee, CRC Press,2021

3. "Hands on Industrial Internet of Things", Giacomo Veneri, Antonio Capasso, Packt Press, 2018

4. "Emerging Technologies for Health and Medicine: Virtual Reality, Augmented Reality, Artificial Intelligence, Internet of Things, Robotics, Industry 4.0", Dac- Nhuong Le Wiley, 2019

5. "Introduction to IoT". S. Misra, A. Mukherjee, and A. Roy Cambridge University Press, 2017

6. "The Internet of Things: Do-It-Yourself at Home Projects for Arduino, Raspberry Pi and Beagle Bone Black", Donald Norris, 2014..

Course Code	SMART CITIES	L	T	P	C
CSPE3XV67		3	0	0	3

COURSE OBJECTIVES:

1. To understand the concept of a smart city and associated challenges.
2. To understand the latest technologies used in intelligent building.
3. To understand the process of planning and drafting a plan for a smart city.
4. To understand the importance of different smart systems.
5. To understand technologies, infrastructure, and the concept of planning and the latest methodology.

UNIT I INTRODUCTION TO SMART CITIES **6**

Introduction to city planning: Concept, Principle stakeholders, key trends in smart cities developments.

UNIT II SMART CITIES PLANNING AND DEVELOPMENT **6**

Understanding smart cities, Dimension of smart cities, Global Standards and performance benchmarks, Practice codes, Smart city planning and development, Financing smart cities development, Governance of smart cities.

UNIT III PROJECT MANAGEMENT IN SMART CITIES **6**

Phases, Stages of project and work break down Structure, Project organization structure, Planning, Scheduling and CPM, Project cost analysis, resource allocation & leveling, Line of balancing technique, Project monitoring and control, Project risk management.

UNIT IV GREEN BUILDING IN SMART CITIES **6**

Introduction to green buildings, Rating system, Energy saving system

UNIT V CASE STUDY **6**

Smart environment, smart streetlight and smart water & waste management, Smart Road & Traffic (Live & Connected roads), Smart Parking (Connected Parking)

TOTAL NUMBER OF PERIODS INCLUDING LAB: 60

SAMPLE LIST OF EXPERIMENTS   NUMBER OF PRACTICAL PERIODS: 30

Prepare a comprehensive project report by analysing the following aspects of a smart city.

1. Analyze key stakeholders and trends in smart city development.
2. Investigate global standards and benchmarks in smart city planning.
3. Implement project management phases and work breakdown structures for smart city projects.
4. Explore green building concepts and energy-saving systems in smart cities.
5. Conduct a case study on smart environment initiatives in urban areas.
6. Evaluate the implementation of smart streetlight and water/waste management systems.

7. Study live and connected road systems for smart traffic management.
8. Implement smart parking solutions using connected technologies.
9. Analyze the financing and governance structures of smart city projects.
10. Develop a comprehensive smart city plan integrating various aspects of planning, development, and management.

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Understand the concept of a smart city and associated challenges.
CO2   Understand the latest technologies used in intelligent building.
CO3   Understand the process of planning and drafting a plan for a smart city.
CO4   Understand the importance of different smart systems.
CO5   Understand technologies, infrastructure, and the concept of planning and the latest methodology.

TEXTBOOKS:

1. "Designing, Developing, and Facilitating Smart Cities Urban Design to IoT Solutions", Vangelis Angelakis Springer, 2019
2. "Introduction to IoT", S. Misra, A. Mukherjee, and A. Roy, Cambridge University Press, 2018

REFERENCES:

1. Jo Beall (1997); "A city for all: valuing differences and working with diversity"; Zed books limited, London (ISBN: 1-85649-477-2)
2. UN-Habitat; "Inclusive and sustainable urban planning: a guide for municipalities"; Volume3: Urban Development Planning (2007); United Nations Human Settlements Programme (ISBN: 978- 92-1-132024-4)
3. Arup Mitra; "Insights into inclusive growth, employment and wellbeing in India"; Springer (2013), New Delhi (ISBN: 978-81-322-0655-2)
4. William J. V. Neill (2004); "Urban Planning and cultural identity"; Routledge, London (ISBN:0- 415-19747-3)
5. John S. Pipkin, Mark E. La Gory, Judith R. Balu (Editors); "Remaking the city: Social science perspective on urban design"; State University of New York Press, Albany (ISBN:0-87395-678-8)
6. Giffinger, Rudolf; Christian Fertner; Hans Kramar; Robert Kalasek; Nataša Pichler-Milanovic;Evert Meijers (2007). "Smart cities – Ranking of European medium-sized cities". Smart Cities.Vienna: Centre of Regional Science
7. "Draft Concept Note on Smart City Scheme". Government of India - Ministry of Urban Development

Course Code	ADVANCED INTELLIGENT SYSTEM	L	T	P	C
CSPE3XV68		2	0	2	3

COURSE OBJECTIVES:

1. Introduce students to the concepts of machine learning and deep learning and their significance in developing intelligent systems.

2. Provide hands-on experience with Python programming for implementing machine learning algorithms such as linear regression, polynomial regression, clustering, and classification.

3. Explore emerging trends in hardware architectures for deep learning, including quantization, precision reduction, and hardware-software co-design.

4. Enable students to develop Python applications specifically for deep learning tasks, focusing on CNN and YOLO algorithms.

5. Engage students in case studies to apply their knowledge and skills in developing intelligent systems for various domains, including power systems, smart energy, motor control, and Industry 4.0 and Industry 5.0 applications.

UNIT I INTELLIGENT SYSTEMS AND PYTHON PROGRAMMING   **6**

Introduction to Machine Learning and Deep Learning - Performance Improvement with Machine Learning - Building Intelligent Systems - Introduction to Python -Python Programming

UNIT II PYTHON FOR ML   **6**

Python Application of Linear Regression and Polynomial Regression using SciPy - Interpolation, Overfitting and Underfitting concepts & examples using SciPy - Clustering and Classification usingPython.

UNIT III EMERGING TRENDS IN HARDWARE ARCHITECTURES FOR DEEP LEARNING   **6**

Quantization and Precision Reduction Techniques - Hardware aware neural Architecture. Hardware-software co-design for deep learning systems Memory hierarchy and cache optimization for deep learning Parallelization and distributed training of deep learning models Energy-efficient deep learning hardware architectures Hardware acceleration for specific deep learning applications(e.g., natural language processing, computer vision)

UNIT IV PYTHON FOR DL   **6**

Python Applications for DL - Python for CNN and YOLO

UNIT V CASE STUDIES   **6**

Development of Intelligent System for Power system protection - Smart Energy - IOE-Motor control - BMS - Intelligent systems for Industry 4.0 and Industry 5.0.

TOTAL NUMBER OF PERIODS INCLUDING LAB: 60

SAMPLE LIST OF EXPERIMENTS   NUMBER OF PRACTICAL PERIODS: 30

1. Implement linear regression and polynomial regression in Python using SciPy, analyze concepts like interpolation, overfitting, and underfitting, then explore clustering and classification algorithms.

2. Investigate quantization and precision reduction techniques in hardware architectures

for deep learning, focusing on hardware-software co-design and memory hierarchy optimization.

3. Develop Python applications for deep learning, specifically for Convolutional Neural Networks (CNN) and You Only Look Once (YOLO) algorithms.

4. Create intelligent systems for various applications such as power system protection, smart energy management, motor control, and building management systems (BMS), exploring Industry 4.0 and Industry 5.0 scenarios.

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Able to gain proficiency in the Python programming language and learn how to apply it in thecontext of intelligent system

CO2   Able to learn Python libraries such as NumPy, Pandas, and scikit-learn to preprocess data, build and train Machine Learning models, and evaluate their performance

CO3   Able to learn Deep Learning libraries such as TensorFlow or PyTorch to build, train, and evaluate Deep Learning models for tasks such as image classification, natural language processing, and computer vision.

CO4   Able to learn hardware components, such as processors, memory, and accelerators, and how they are integrated.

CO5   Able to learn intelligent systems implementations, examine their design choices, evaluate their performance, and understand the challenges.

TEXT CUM REFERENCE BOOKS:

1. "Intelligent Systems: Principles, Paradigms, and Pragmatics" by Rajendra P. Srivastava (Published in 2013)

2. "Intelligent Systems: A Modern Approach" by Thomas Bäck, David B. Fogel, and ZbigniewMichalewicz (Published in 2000)

3. "Intelligent Systems: Modeling, Optimization, and Control" by Grzegorz Bocewicz and KonradJackowski (Published in 2016)

4. "Intelligent Systems: Architecture, Design, and Control" by Janos Sztipanovits and Gabor Karsai(Published in 2018)

5. "Intelligent Systems: Concepts and Applications" by Veera M. Boddu (Published in 2017)

HIGH END COMPUTING

Course Code	PARALLEL PROCESSING	L	T	P	C
CSPE3XV71		3	0	0	3

COURSE OBJECTIVES:

1. To study the scalability and clustering issues and the technology necessary for them.
2. To understand the technologies enabling parallel computing.
3. To study the different types of interconnection networks.
4. To study the different parallel programming models.
5. To study the software support needed for shared memory programming

UNIT I INTRODUCTION   **9**

Introduction, Parallel Processing – Shared Memory Multiprocessing – Distributed Shared Memory – Message Passing Parallel Computers.

UNIT II PROCESSES & SHARED MEMORY PROGRAMMING   **9**

Processes - Shared Memory Programming – General Model Of Shared Memory Programming – Forking-Creating Processes – Joining Processes - Process Model Under UNIX.

UNIT III BASIC PARALLEL PROGRAMMING TECHNIQUES: **9**

Loop Splitting – Ideal Speedup – Spin-Locks, Contention And Self- Scheduling. Scheduling : Loop Scheduling – Variations On Loop Scheduling – Self- Scheduling – Variations On Self-Scheduling – Indirect Scheduling – Block Scheduling.

UNIT IV THREAD-BASED IMPLEMENTATION   **9**

Thread Management – The POSIX Thread Application Programmer Interface- Synchronization Primitives in POSIX- Example With Threads – Attributes Of Threads – Mutual Exclusion With Threads – Mutex Usage Of Threads – Thread Implementation –Events And Condition Variables – Deviation Computation With Threads – Java Threads.

UNIT V ALGORITHMS FOR PARALLEL MACHINES   **9**

Models Of Computation – Analysis Of Parallel Algorithms – Prefix Computation – Histogram Computation – Parallel Reduction – Sorting Networks - Matrix Multiplication

          45

TOTAL PERIODS :

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Understand the basics concepts of Parallel Processing.

CO2   Apply the concepts of processes and shared memory programming.

CO3   Use basic parallel programming techniques.

CO4   Implementations of thread based methods.

CO5   Understand parallel algorithms for tightly coupled and loosely coupled parallel systems for various applications

TEXT BOOKS:

1 Introduction To Parallel Programming - By Steven Brawer.
2 Introduction to Parallel Computing, Ananth Grama, Anshul Gupta, George Karypis, Vipin Kumar, By Pearson Publication.
3 Introduction To Parallel Processing – By M.Sasikumar, Dinesh Shikhare And P. Ravi Prakash.
4 M.Sasikumar, D.Shikhare and P. RaviPrakash, "Introduction to Parallel processing". PHI 2006

REFERENCES:

1 Hwang. K, "Advanced computer Architecture", Parallelism, Scalability, Programmability, Tata McGraw Hill, 3rd Edition, 1993.
2 Tanenbaum A.S, "Distributed Operating Systems", Pearson Education Asia, 2002.
3 DezsoSima, Terence Fountain, Peter Kacsuk, "Advanced Computer Architectures", Pearson Education, 2007.
4 V.Rajaraman and C.Siva Ram Murthy, "Parallel Computers Architecture and Programming", PHI, 2000.
5 Quinn, M.J., "Designing Efficient Algorithms for Parallel Computers", McGraw – Hill, 2003.
6 Culler, D.E., "Parallel Computer Architecture, A Hardware – Software approach", Morgan Kaufmann Publisher, 1998.

Course Code	HIGH-PERFORMANCE COMPUTING	L	T	P	C
CSPE3XV72		3	0	0	3

COURSE OBJECTIVES:

1. To Study various computing technology architectures.
2. To know Emerging trends in computing technology.
3. To highlight the advantage of deploying computing technology

UNIT I CLUSTER COMPUTING AND ITS ARCHITECTURE: **9**

Ease of Computing, Scalable Parallel Computer Architecture , Towards Low Cost Parallel Computing & Motivation, Windows opportunity, A Cluster Computer And Its Architecture, Cluster Classification, Commodity Components for Clusters Network Services/Communication SW, Cluster Middleware and Single Systems Image, Resource management & Scheduling (RMS).

UNIT II CLUSTER SETUP AND ADMINISTRATION: **9**

Introduction, Setting up the cluster, Security, System Monitoring, and System Tuning. Introduction to Grid and its Evolution:, Introduction to Grid and its Evolution, Beginning of the Grid, Building blocks of Grid, Grid Application and Grid Middleware, Evolution of
the Grid: First, Second & Third Generation.

UNIT III CLOUD DEFINITION AND MANAGEMENT: **9**

Introduction to Cloud Computing, Defining Clouds, Cloud Providers, Consuming Cloud Services, Cloud Models, Iaas, Paas, SaaS, Inside the cloud, Administering cloud services, technical interface, and Cloud resources.

UNIT IV NATURE OF CLOUD **9**

Tradition Data Center, Cost of Cloud Data Center, Scaling computer systems, Cloud workload, Managing data on clouds, Public, private, and hybrid clouds

UNIT V CLOUD ELEMENTS **9**

Infrastructure as a service, Platform as a service, Software as a service.

TOTAL PERIODS: 45

COURSE OUTCOMES:

On completion of this course, the students will be able to:.

CO1 On successful completion of the course, the student will be having the basic knowledge of computing technology.
CO2 Student will be able to understand architecture of computing technology.
CO3 Student will be able to know cloud computing service models.
CO4 Know about emerging trends in computing technology
CO5 Student will be able to know big data and hadoop architecture.

TEXT CUM REFERENCE BOOKS:

1 Ronald Krutz, Cloud Security, Wiley India.
2 Cloud Computing, A Practical Approach, Anthony Velte, Toby Velte, Robert Elsenpeter, McGrawHill.

Course Code	PERVASIVE COMPUTING	L	T	P	C
CSPE3XV73		3	0	0	3

COURSE OBJECTIVES:

1. To understand the characteristics and principles of Pervasive computing and the solutions that are in use
2. To realize the role of wireless protocols in shaping the future Internet
3. To design and implement pervasive applications
4. To give an introduction to the enabling technologies of pervasive computing

UNIT I INTRODUCTION   **9**

Pervasive Computing- Principles, Characteristics- interaction transparency, context aware, automated experience capture. Architecture for pervasive computing- Pervasive devices- embedded controls.- smart sensors and actuators -Context communication and access services.

UNIT II PROTOCOLS   **9**

Open protocols- Service discovery technologies- SDP, Jini, SLP, UpnP protocols–data synchronization- SyncML framework - Context aware mobile services - Context aware sensor networks, addressing and communications- Context aware security.

UNIT III TECHNOLOGIES   **9**

Past, Present and Future-Device Technology-Device Connectivity-Web application Concepts-WAP and Beyond-Voice Technologies-Personal Digital Assistants.

UNIT IV ARCHITECTURE   **9**

Server side programming in Java-Pervasive Web application Architecture-Example Application- Access via PCs-Access via WAP-Access via PDA and Voice.

UNIT V EXAMPLES   **9**

Smart Tokens, Heating Ventilation and Air Conditioning, Set Top Boxes, Appliances and Home Networking, Residential Gateway, Automotive Computing, On Board Computing Systems, In Vehicle networks, Entertainment Systems

TOTAL PERIODS : 45

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Outline the basic problems, performance requirements of pervasive computing applications, and the trends of pervasive computing and its impacts on future computing applications and
      society

CO2   Analyze and compare the performance of different data dissemination techniques and algorithms for mobile real-time applications

CO3   Analyze the performance of different sensor data management and routing algorithms for sensor networks

CO4   Develop an attitude to propose solutions with comparisons for problems related to pervasive computing system through investigation

REFERENCES:

1   Seng Loke, Context-Aware Computing Pervasive Systems, Auerbach Pub., New York, 2007.
2   Uwe Hansmann etl , Pervasive Computing, Springer, New York,2001.
3   Jochen Burkhardt, , Stefan Hepper, Klaus Rindtorff, Thomas Schaeck "Pervasive Computing-Technology and Architecture of Mobile Internet Application",Pearson Education, sixth Edition 2009.

Course Code	PICO COMPUTING	L	T	P	C
CSPE3XV74		3	0	0	3

COURSE OBJECTIVES:

1. Learn Nano computing challenges
2. Be familiar with the imperfections
3. Be exposed to reliability evaluation strategies
4. Learn Nanoscale quantum computing
5. Understand Molecular Computing and Optimal Computing

UNIT I NANO COMPUTING-PROSPECTS AND CHALLENGES   **9**

Introduction - History of Computing - Nanocomputing - Quantum Computers – Nanocomputing
Technologies - Nano Information Processing - Prospects and Challenges - Physics of
Nanocomputing: Digital Signals and Gates - Silicon Nanoelectronics - Carbon Nanotube
Electronics – Carbon Nanotube Field-effect Transistors – Nanolithography.

UNIT II NANOCOMPUTING WITH IMPERFECTIONS   **9**

Introduction - Nanocomputing in the Presence of Defects and Faults - Defect Tolerance –
Towards Quadrillion Transistor Logic Systems.

UNIT III RELIABILITY OF NANOCOMPUTING   **9**

Markov Random Fields - Reliability Evaluation Strategies - NANOLAB - NANOPRISM – Reliable
Manufacturing and Behavior from Law of Large Numbers.

UNIT IV NANOSCALE QUANTUM COMPUTING   **9**

Quantum Computers - Hardware Challenges to Large Quantum Computers - Fabrication,
Test, and Architectural Challenges - Quantum-dot Cellular Automata (QCA)
- Computing with QCA – QCA Clocking - QCA Design Rules.

UNIT V QCA DESIGNER SOFTWARE AND QCA IMPLEMENTATION   **9**

Basic QCA Circuits using QCA Designer - QCA Implementation - Molecular and Optical
Computing: Molecular Computing - Optimal Computing - Ultrafast Pulse Shaping and Tb/sec
Data Speeds.

TOTAL PERIODS : 45

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   To design the basic components in Nano Computing.
CO2   **T**o construct the Logic Devices
CO3   To understand the reliability evaluation strategies.
CO4   To analyze Quantum Computing and Challenges.
CO5   To analyze the principles of QCA Design and Implementation.

TEXTBOOKS

1. Sahni V. and Goswami D., Nano Computing, McGraw Hill Education Asia Ltd. (2008), ISBN
   (13): 978007024892.

REFERENCES:

1. Sandeep K. Shukla and R. Iris Bahar., Nano, Quantum and Molecular Computing, Kluwer Academic Publishers 2004, ISBN: 1402080670
2. Sahni V, Quantum Computing, McGraw Hill Education Asia Ltd. 2007
3. Jean-Baptiste Waldner, Nanocomputers and Swarm Intelligence, John Wiley & Sons, Inc. 2008, ISBN (13): 978-1848210097.

Course Code	FOG AND EDGE COMPUTING	L	T	P	C
CSPE3XV57		2	0	2	3

COURSE OBJECTIVES:

6. To enhance real-time data processing and analytics at the network edge.
7. To optimize resource utilization and reduce latency in fog computing environments.
8. To improve scalability and flexibility for edge devices and applications.
9. To Enhance security and privacy for data processing at the edge.
10. To Facilitate seamless integration of fog and edge computing with cloud services.

UNIT I INTRODUCTION TO FOG AND EDGE COMPUTING   6

Definition and Fundamentals: Overview of Fog and Edge Computing, Key concepts and characteristics, Motivations and Advantages: Reasons for adopting Fog and Edge Computing, Comparative advantages over traditional Cloud Computing, Use Cases and Applications: Challenges and Limitations: Security concerns, privacy issues, and compliance, Comparison with Cloud Computing: Contrasting characteristics and use cases, Complementary roles in a hybrid computing environment.

UNIT II ARCHITECTURES AND COMPONENTS   6

Edge Computing Architecture Models: Design principles and characteristics, Hierarchical vs. flat architectures. Fog Nodes and Infrastructure: Role and capabilities of Fog nodes, Infrastructure components supporting Edge Computing. Edge Device Types and Capabilities: Categorization of Edge devices (e.g., gateways, sensors, actuators), Capabilities and limitations of different device types. Edge-to-Cloud Communication Models; Middleware and Software Frameworks: Middleware solutions for Edge Computing.

UNIT III SECURITY AND PRIVACY IN EDGE COMPUTING   6

Security Challenges in Edge Computing: Identifying security threats in Edge Computing, Risks associated with decentralized architectures. Methods for secure access control: Role of authentication in ensuring device integrity, Data Encryption in Edge Environments: Encryption strategies for data at rest and in transit. Privacy Concerns and Regulatory Compliance; Risk Management and Incident Response.

UNIT IV EDGE APPLICATION DEVELOPMENT   6

Programming Models and Frameworks: Overview of programming models for Edge Computing, Exploration of popular frameworks for Edge application development, Integration with IoT Devices:
Techniques for integrating Edge Computing with Internet of Things (IoT) devices. Edge Application Deployment Strategies; Edge-to-Cloud Communication Patterns; Optimizing Edge Application Performance: Performance considerations in Edge Computing, Strategies for optimizing resource usage and responsiveness.

UNIT V PERFORMANCE OPTIMIZATION AND FUTURE TRENDS   6

Performance Metrics and Optimization: Metrics for evaluating performance in Edge Computing
Strategies for optimizing Edge applications; Energy-Efficient Edge Architectures: Techniques for optimizing energy consumption, Green computing practices in Edge environments, Real-time Processing and Analytics: Approaches for real-time data processing at the Edge, Analytics capabilities in decentralized architectures. Scalability and Flexibility in Edge

Architectures; Emerging Trends in Fog and Edge Computing.

TOTAL NUMBER OF PERIODS INCLUDING LAB: 60

SAMPLE LIST OF EXPERIMENTS   NUMBER OF PRACTICAL PERIODS: 30

11. Design and deploy a fog computing architecture for a smart city application.
12. Optimize data processing algorithms for edge devices in a real-time monitoring system.
13. Evaluate resource utilization in a fog computing environment using simulation tools.
14. Implement security protocols for edge devices to protect sensitive data.
15. Develop a scalable edge computing solution for IoT devices in a manufacturing setting.
16. Integrate fog computing with cloud services to enhance overall system efficiency.
17. Perform a hands-on deployment of edge computing nodes in a network infrastructure.
18. Design and implement a fault-tolerant edge computing solution for critical applications.
19. Conduct performance testing to assess the latency reduction achieved through edge computing.
20. Create a comprehensive case study on the successful integration of fog and edge computing in a specific industry.

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Proficiently deploy and manage fog and edge computing solutions in diverse environments.
CO2   Demonstrate expertise in optimizing data processing and analytics at the network edge.
CO3   Evaluate and enhance resource efficiency for fog computing deployments.
CO4   Implement robust security measures for safeguarding edge computing data.
CO5   Successfully integrate fog and edge computing technologies to address real-world challenges.

TEXT& REFERENCE BOOKS

1. "Fog and Edge Computing: Principles and Paradigms" by Rajkumar Buyya and Amir Vahid Dastjerdi (Wiley, 1st Edition, 2018),
2. "Edge Computing: A Primer" by Shiwen Mao and Ying-Chang Liang (Wiley, 1st Edition, 2019),
3. "Fog Computing in the Internet of Things: Intelligence at the Edge" by Yogesh Simmhan, Nalini Venkatasubramanian, and Viktor K. Prasanna (Springer, 1st Edition, 2018),
4. "Fog Computing: Concepts, Frameworks and Technologies" by Flavio Bonomi, Rodolfo Milito, Jiang Zhu, and Sateesh Addepalli (Springer, 1st Edition, 2018),
5. "Edge Computing – The Dawn of Industry 4.0 Revolution" by Satya Prakash Ghrera (CRC Press, 1st Edition, 2020)
6. "Fog Computing: Towards Minimizing Latency and Improving QoE in the Internet of Things" by Mohammad Mehedi Hassan, Saad B. Qaisar, and Abdelgadir Mohammed Ahmed (Springer, 1st Edition, 2018).

Course Code	BLOCKCHAIN TECHNOLOGY AND CLOUD COMPUTING	L	T	P	C
CSPE3XV58		2	0	2	3

COURSE OBJECTIVES:

1. Introduce students to the core principles, mechanisms, and applications of Blockchain Technology and CloudComputing.

2. Foster a solid theoretical foundation in Blockchain mechanisms, cryptographic principles, cloud architectures,and enabling technologies.

3. Provide hands-on experience through lab sessions, experiments, and projects to apply theoretical conceptsinto practical scenarios.

4. Enable students to integrate the principles and practices of Blockchain and Cloud Computing to developinnovative and efficient solutions.

5. Empower students to analyse real-world use cases and explore the potential of utilizing Blockchain and CloudComputing in various applications.

UNIT I INTRODUCTION TO BLOCKCHAIN AND CLOUD COMPUTING   6

Blockchain Technology Mechanisms & Networks: Overview, Blockchain Networks, Decentralization. **Transactions, Blocks, P2P Systems:** Structure of Transactions, Concept of Blocks, Peer-to-Peer (P2P) Systems. **Keys, Digital Signatures, Hashing:** Cryptographic Keys, Digital Signatures in Blockchain Transactions, Hash Functions and Applications. **Cloud Computing - Definition, Evolution, Characteristics**: Introduction to Cloud Computing, Characteristics, Evolution of Cloud. **Principles of Parallel Computing, Elasticity:** Principles of Parallel Computing in Cloud, Elasticity and Scalability in Cloud Environments, Techniques for Parallelism. **On-demand Provisioning:** Understanding On-demand Provisioning in Cloud, Resource Allocation and De-allocation, Implementing Scalable Resources on Demand.

UNIT II CYPTOCURRENCY AND CLOUD ENABLING TECHNOLOGY   6

Bitcoin Overview: Introduction to Bitcoin, Bitcoin Mining Process, Bitcoin Wallets: Types and Functions, Ethereum Virtual Machine (EVM). **Blockchain & Cryptocurrency:** Consensus Mechanisms in Blockchain, Smart Contracts and Applications, Decentralization and its Significance, Impact of Blockchain Technology on Cryptocurrency. **Understanding Cloud Basics:** Service-Oriented Architecture (SOA), Fundamentals of Virtualization, Types and Levels of Virtualization, Cloud Characteristics and Attributes, Elasticity and Scalability in Cloud Computing. **Principles and Implementation**: Cloud Services and Deployment Models, Cloud Virtualization Tools and Mechanisms. CPU, Memory, and I/O Devices Virtualization, Disaster Recovery in Cloud Environments.

UNIT III ETHEREUM & CLOUD ARCHITECTURE   6

Overview of Ethereum: Ethereum Origins and Core Principles, Smart Contracts, Decentralization. **Ethereum Operations & Transactions:** Ethereum Accounts: Types and Functions, Transaction Process in Ethereum Network, Ethereum Smart Contracts: Structure and Execution. **Understanding Hyperledger:** Introduction to Hyperledger Fabric, Hyperledger Composer: Overview and Functions, Consensus Mechanisms in Hyperledger. **Layered Cloud ArchitectureDesign:** Components of Cloud Architecture, Exploring Layers in Cloud Environments, Architectural Design Challenges. **Cloud Services & Service Models:** Overview of Cloud Services (IaaS, PaaS, SaaS), Characteristics and Applications of Each Service Model, Challenges and Solutions in Service Deployment. **Cloud Storage Solutions:**

Cloud Storage Basics and Principles, Storage-as-a-Service (STaaS) Concept, Analysis of Cloud Storage Providers (e.g., S3).

UNIT IV SOLIDITY PROGRAMMING AND CLOUD SECURITY   6

Solidity Programming Language: Syntax, Types, Variables, Functions, Data Structures in Solidity. **Ethereum Wallet & Smart Contracts:** Overview of Ethereum Wallet, Structure & Working of Smart Contracts. **Basics of Solidity & Smart Contract Structure:** Understanding Basic Solidity Concepts, Components and Structure of Smart Contracts. **Resource Provisioning & Cloud Overview:** Cloud Resource Allocation & Management, Cloud Service Models (IaaS, PaaS, SaaS). **Security Overview in Cloud Environment:** Cloud Security Challenges, Security Governance in Cloud. **IAM, Virtual Machine Security, Security Standards:** Identity & Access Management in Cloud, Ensuring Security in Virtual Machines, Compliance with Security Standards in Cloud Environments.

UNIT V BLOCKCHAIN APPLICATIONS AND ADVANCEMENT OF CLOUD TECHNOLGIES   6

IoT Integration with Blockchain: Introduction to IoT (Internet of Things), Blockchain's Role in IoT Security, IoT Data Management using Blockchain. **Blockchain in Medical Record Management:** Healthcare Data Challenges & Security, Blockchain's Impact on Medical Records, Privacy, Integrity, and Accessibility in Medical Records, Applications. **Domain Name Service (DNS) using Blockchain:** DNS Issues in Centralized Systems, Blockchain-Based Decentralized DNS. **Alt Coins and Their Significance:** Introduction to Alt Coins (Alternative Cryptocurrencies), Role and Market Impact of Alt Coins, Differentiating Factors and Advantages, Future Trends and Adoption of Alt Coins. **Future Trends in Blockchain Technology:** Emerging Trends, Impact on Industries and Potential Evolution. **Hadoop and Big Data Processing in Cloud:** Introduction to Hadoop and MapReduce, Cloud-Based Big Data Analytics, Hadoop's Role in Cloud Computing, Use Cases and Industry Applications. **Google App Engine (GAE) for Cloud Development:** Overview of GAE, Building and Deploying Applications on GAE, Scalability and Maintenance Aspects, GAE's Contribution to Cloud Development. **OpenStack and Cloud Federation:** OpenStack in Cloud Environments, Federation in Cloud Computing, Multi-Cloud Management using OpenStack, Benefits and Challenges of Cloud Federation.

TOTAL NUMBER OF PERIODS INCLUDING LAB: 60

SAMPLE LIST OF EXPERIMENTS   NUMBER OF PRACTICAL PERIODS: 30

1. Simulate a basic blockchain network to understand transactions, blocks, and the structure of peer-to-peer systems, alongside implementing on-demand resource provisioning in cloud computing.

2. Analyze the Bitcoin mining process, types of wallets, and Ethereum Virtual Machine (EVM) while exploring consensus mechanisms and smart contracts' impact on cryptocurrency, and understanding cloud basics such as service-oriented architecture (SOA) and virtualization.

3. Develop smart contracts in Solidity programming language for Ethereum, comprehend Hyperledger Fabric and Hyperledger Composer, and design layered cloud architecture considering cloud services and service models.

4. Implement Solidity programming language features like variables, functions, and data structures, alongside understanding Ethereum wallet and smart contracts structure, and ensuring cloud security through identity & access management (IAM) and compliance with security standards.

5. Explore blockchain applications like IoT integration and medical record management, analyze alternative cryptocurrencies (Alt Coins) and their significance, and investigate advancements in cloud technologies such as Hadoop for big data processing and Google App Engine (GAE) for cloud development.

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Demonstrate a comprehensive understanding of the fundamental principles and mechanisms of BlockchainTechnology and Cloud Computing.

CO2   Apply theoretical knowledge to solve problems related to Blockchain and Cloud Computing through hands-onexperiments and practical sessions.

CO3   Integrate Blockchain Technology and Cloud Computing for the development of innovative solutions,leveraging the combined strengths of both domains.

CO4   Exhibit problem-solving skills by analysing and proposing solutions using Blockchain and Cloud Computingprinciples in diverse scenarios.

CO5   Analyse and evaluate real-world use cases to determine the suitability and potential impact of Blockchain andCloud Computing technologies.

CO6   Develop practical proficiency in executing projects that utilize Blockchain and Cloud Computing for addressingcontemporary challenges.

TEXTBOOKS:
1. Daniel Drescher, "Blockchain Basics: A Non-Technical Introduction in 25 Steps", 1st Edition, 2017
2. Andreas M. Antonopoulos, "Mastering Bitcoin: Unlocking Digital Cryptocurrencies", 2nd Edition, 2017
3. Stephen Grider, "Ethereum Programming", 1st Edition, 2020.
4. Thomas Erl, Ricardo Puttini, and Zaigham Mahmood, "Cloud Computing: Concepts, Technology &Architecture", 2nd edition, 2013.
5. Michael J. Kavis, "Architecting the Cloud: Design Decisions for Cloud Computing Service Models", 1st edition,2014.

REFERENCES:

1. Don Tapscott and Alex Tapscott, "Blockchain Revolution: How the Technology Behind Bitcoin and OtherCryptocurrencies is Changing the World", 2nd edition, 2018.
2. Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller, and Steven Goldfeder, "Bitcoin andCryptocurrency Technologies: A Comprehensive Introduction", 1st edition, 2016.
3. Judith Hurwitz, Robin Bloor, Marcia Kaufman, and Fern Halper, "Cloud Computing for Dummies", 1st edition,2009.
4. Tim Mather, Subra Kumaraswamy, and Shahed Latif, "Cloud Security and Privacy: An Enterprise Perspectiveon Risks and Compliance", 1st edition, 2009.

APPENDIX B: OPEN ELECTIVES

S. NO.	COURSE CODE	COURSE TITLE	CATEGORY	PERIODS PER WEEK			CREDITS
				L	T	P	
1.	CSOE3001	Introduction to R Programming	OEC	3	0	0	3
2.	CSOE3002	User Interface Design	OEC	3	0	0	3
3.	CSOE3003	Internet Marketing and E-Commerce	OEC	3	0	0	3
4.	CSOE3004	Cognitive Science	OEC	3	0	0	3
5.	CSOE3005	Emotional Intelligence	OEC	3	0	0	3
6.	CSOE3006	Design Thinking	OEC	3	0	0	3
7.	CSOE3007	Electronic Waste Management - Issues and Challenges	OEC	3	0	0	3
8.	CSOE3008	Fundamentals of GIS	OEC	3	0	0	3
9.	CSOE3009	Bioinformatics	OEC	3	0	0	3
10.	CSOE3010	Business Strategy	OEC	3	0	0	3
11.	CSOE3011	Introduction to Cybersecurity	OEC	3	0	0	3
12.	CSOE3012	Green Computing	OEC	3	0	0	3
13.	CSOE3013	E – Business Management	OEC	3	0	0	3
14.	CSOE3014	Multicore Architecture and Programming	OEC	3	0	0	3
15.	CSOE3015	Augmented Reality and Virtual Reality	OEC	3	0	0	3
16.	CSOE3016	Internet of Things	OEC	3	0	0	3
17.	CSOE3017	Cyber Law	OEC	3	0	0	3

Course Code	**INTRODUCTION TO R PROGRAMMING**	L	T	P	C
CSOE3001		3	0	0	3

COURSE OBJECTIVES:

6. Develop proficiency in programming using the R language.
7. Acquire skills for data manipulation, visualization, and statistical analysis.
8. Understand advanced programming concepts and techniques in R.
9. Apply R programming skills to solve real-world data science problems.
10. Foster critical thinking and problem-solving abilities through hands-on projects and assignments.

UNIT I INTRODUCTION TO R   **9**

Introduction To The R Language, Understand R syntax, data types, and data structures., Learn how to install and set up R and RStudio. Explore basic operations and functions in R.

UNIT II DATA MANIPULATION AND VISUALIZATION   **9**

Teach students how to manipulate and visualize data using R., Use the dplyr and tidyr packages for data manipulation., Create basic and advanced plots using ggplot2.
Understand principles of effective data visualization.

UNIT III STATISTICAL ANALYSIS WITH R   **9**

Enable students to perform statistical analysis and hypothesis testing in R. - Conduct descriptive statistics and exploratory data analysis, Perform inferential statistics and hypothesis tests., Implement linear regression and other statistical models.

UNIT IV ADVANCED R PROGRAMMING TECHNIQUES   **9**

Introduce students to advanced programming techniques in R., Learn how to write custom functions and control structures., Understand object-oriented programming in R. Explore techniques for debugging and error handling.

UNIT V DATA SCIENCE APPLICATIONS WITH R   **9**

Apply R programming skills to solve real-world data science problems. Work on data analysis projects using publicly available datasets. Apply machine learning algorithms for predictive modelling. Learn best practices for reproducible research and project management in R.

NUMBER OF THEORY PERIODS: 45

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Demonstrate proficiency in writing R code for data manipulation, visualization, and statistical analysis.

CO2   Apply statistical techniques and hypothesis testing methods to analyze data and draw conclusions.

CO3   Implement advanced programming concepts such as custom function writing and object-oriented programming in R.

CO4   Successfully complete R analysis projects, including data cleaning, exploration, modelling, and visualization.

CO5   Develop critical thinking and problem-solving skills by tackling real-world data science challenges and effectively communicating results.

TEXTBOOKS:

1. Roger D. Pengm, "R Programming for Data Science", O'Reilly Media, 2016.

REFERENCES:

1. "The Art of R Programming: A Tour of Statistical Software Design" by Norman Matloff
2. "R Cookbook: Proven Recipes for Data Analysis, Statistics, and Graphics" by Winston Chang
3. "Advanced R" by Hadley Wickham
4. "Introductory Statistics with R" by Peter Dalgaard
5. "Hands-On Programming with R" by Garrett Grolemund

Course Code	USER INTERFACE DESIGN	L	T	P	C
CSOE3002		3	0	0	3

COURSE OBJECTIVES:

1. To understand the characteristics and principles of human-computer interaction (HCI), including graphical user interfaces and web interfaces.
2. To explore the user interface design process, focusing on usability and incorporating human characteristics into design considerations.
3. To examine the components and characteristics of windows in graphical user interfaces, including presentation styles, types, and management.
4. To learn about multimedia elements in user interfaces, such as text, images, icons, and multimedia content, and their effective utilization.
5. To gain knowledge of window layout design, including the creation of prototypes, testing methods, information search techniques, and visualization in HCI.

UNIT I INTRODUCTION   9

Human–Computer Interface – Characteristics Of Graphics Interface –Direct Manipulation Graphical System – Web User Interface –Popularity –Characteristic & Principles.

UNIT II HUMAN COMPUTER INTERACTION   9

User Interface Design Process – Obstacles –Usability –Human Characteristics In Design – Human Interaction Speed –Business Functions –Requirement Analysis – Direct – Indirect Methods – Basic Business Functions – Design Standards – System Timings – Human Consideration In Screen Design – Structures Of Menus – Functions Of Menus– Contents Of Menu– Formatting – Phrasing The Menu – Selecting Menu Choice– Navigating Menus– Graphical Menus.

UNIT III WINDOWS   9

Characteristics– Components– Presentation Styles– Types– Managements– Organizations– Operations– Web Systems– Device– Based Controls Characteristics– Screen – Based Controls – Operate Control – Text Boxes– Selection Control– Combination Control– Custom Control– Presentation Control.

UNIT IV MULTIMEDIA   9

Text For Web Pages – Effective Feedback– Guidance & Assistance– Internationalization– Accessibility– Icons– Image– Multimedia – Colouring.

UNIT V WINDOWS LAYOUT   9

Prototypes – Kinds Of Tests – Retest – Information Search – Visualization – Hypermedia – WWW– Software Tools.

NUMBER OF THEORY PERIODS :

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Students will be able to analyse and apply the characteristics and principles of HCI to design effective graphical and web interfaces.

CO2   Students will demonstrate proficiency in the user interface design process, including

requirement analysis, design standards, and consideration of human factors.

CO3 Students will be able to identify and utilize various components of windows in graphical user interfaces, adapting presentation styles and controls based on user needs.

CO4 Students will develop skills in incorporating multimedia elements, such as text, images, and multimedia content, into user interfaces to enhance user experience.

CO5 Students will be able to design effective window layouts, including creating prototypes, conducting tests, and utilizing visualization techniques to improve HCI outcomes.

TEXTBOOKS:

1. Wilbent. O. Galitz ,"The Essential Guide To User Interface Design", John Wiley& Sons, 2001.

2. Ben Sheiderman, "Design The User Interface", Pearson Education, 1998.

REFERENCES:

1. Alan Cooper, "The Essential Of User Interface Design", Wiley – Dream Tech Ltd., 2002.

Course Code	INTERNET MARKETING AND E-	L	T	P	C
CSOE3003	COMMERCE	3	0	0	3

COURSE OBJECTIVES:

1. To comprehend the historical development of e-commerce and its significance in the Indian business context.
2. To explore the various aspects of e-marketing and its integration with traditional marketing principles.
3. To examine the critical aspects of e-security and legal issues surrounding e-commerce.
4. To comprehend the different types of digital payment systems and their functionalities.
5. To understand the concepts and technologies underlying mobile commerce and human resource portals.

UNIT I INTRODUCTION 9

History of E-commerce and Indian Business Context: E-Commerce –Emergence of the Internet – Emergence of the WWW – Advantages of E-Commerce – Transition to E-Commerce in India – The Internet and India – E-transition Challenges for Indian Corporate. Business Models for E-commerce: Business Model – E-business Models Based on the Relationship of Transaction Parties - E-business Models Based on the Relationship of Transaction Types.

UNIT II WWW AND E-MARKETING 9

Enabling Technologies of the World Wide Web: World Wide Web – Internet Client-Server Applications –Networks and Internets – Software Agents – Internet Standards and Specifications – ISP. e-Marketing: Traditional Marketing – Identifying Web Presence Goals – Online Marketing – E-advertising – E-branding.

UNIT III E-SECURITY 9

E-Security: Information system Security – Security on the Internet – E-business Risk Management Issues – Information Security Environment in India. Legal and Ethical Issues : Cyberstalking – Privacy is at Risk in the Internet Age – Phishing – Application Fraud – Skimming – Copyright – Internet Gambling – Threats to Children.

UNIT IV DIGITAL PAYMENTS 9

e-Payment Systems: Main Concerns in Internet Banking – Digital Payment Requirements – Digital Token-based e-payment Systems – Classification of New Payment Systems – Properties of Electronic Cash – Cheque Payment Systems on the Internet – Risk and e-Payment Systems – Designing e-payment Systems – Digital Signature – Online Financial Services in India - Online Stock Trading.

UNIT V MOBILE COMMERCE 9

Information systems for Mobile Commerce: What is Mobile Commerce? – Wireless Applications – Cellular Network – Wireless Spectrum – Technologies for Mobile Commerce – Wireless Technologies –Different Generations in Wireless Communication – Security Issues Pertaining to Cellular Technology. Portals for E-Business: Portals – Human Resource Management – Various HRIS Modules.

NUMBER OF THEORY PERIODS: 45

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1 To trace the emergence of the Internet, the World Wide Web, and analyze the advantages of e-commerce.

CO2 To understand web presence goals, online marketing techniques, e-advertising, and e-branding strategies.

CO3 To learn about information system security, e-business risk management, and legal and ethical concerns such as cyberstalking, privacy infringement, phishing, and copyright issues.

CO4 To classify and evaluate digital payment systems, understand the properties of electronic cash, assess the risks associated with e-payment systems, and analyze the design considerations for implementing secure e-payment systems.

CO5 To gain knowledge about mobile commerce, wireless technologies, cellular networks, and security issues related to cellular technology

TEXTBOOKS:

1. P.T.Joseph, S.J., "E-Commerce - An Indian Perspective", PHI 2012, 4th Edition.

REFERENCES:

1. David Whiteley , "E-Commerce Strategy, Technologies and Applications", Tata McGraw Hill, 2001.
2. 2. Ravi Kalakota, Andrew B Whinston, "Frontiers of Electronic Commerce", Pearson 2006, 12th Impression.

WEB REFERENCES:

1. https://www.docsity.com/en/e-commerce-notes-pdf-lecture-notes-universitylevel/2484734/
2. https://magnetoitsolutions.com/blog/advantages-and-disadvantages-of-ecommerce
3. https://www.researchgate.net/publication/320547139ECommerce_Merits_and_Demerits_A_Review_Paper

Course Code	COGNITIVE SCIENCE	L	T	P	C
CSOE3004		3	0	0	3

COURSE OBJECTIVES:

1. To know the theoretical background of cognition.
2. To understand the link between cognition and computational intelligence.
3. To explore probabilistic programming language.
4. To study the computational inference models of cognition.
5. To study the computational learning models of cognition.

UNIT I PHILOSOPHY, PSYCHOLOGY AND NEUROSCIENCE **9**

Philosophy: Mental-physical Relation – From Materialism to Mental Science – Logic and the Sciences of the Mind – Psychology: Place of Psychology within Cognitive Science – Science of Information Processing –Cognitive Neuroscience – Perception – Decision – Learning and Memory – Language Understanding and Processing.

UNIT II COMPUTATIONAL INTELLIGENCE **9**

Machines and Cognition – Artificial Intelligence – Architectures of Cognition – Knowledge Based Systems – Logical Representation and Reasoning – Logical Decision Making –Learning – Language – Vision.

UNIT III PROBABILISTIC PROGRAMMING LANGUAGE **9**

WebPPL Language – Syntax – Using Javascript Libraries – Manipulating probability types and distributions – Finding Inference – Exploring random computation – Coroutines: Functions that receive continuations –Enumeration

UNIT IV INFERENCE MODELS OF COGNITION **9**

Generative Models – Conditioning – Causal and statistical dependence – Conditional dependence – Data Analysis – Algorithms for Inference.

UNIT V LEARNING MODELS OF COGNITION **9**

Learning as Conditional Inference – Learning with a Language of Thought – Hierarchical Models– Learning (Deep) Continuous Functions – Mixture Models.

NUMBER OF THEORY PERIODS :45

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Understand the underlying theory behind cognition.
CO2   Connect to the cognition elements computationally.
CO3   Implement mathematical functions through WebPPL.
CO4   Develop applications using cognitive inference model.
CO5   Develop applications using cognitive learning model.

TEXTBOOKS:

1. Vijay V Raghavan,Venkat N.Gudivada, VenuGovindaraju, C.R. Rao, Cognitive Computing:
2. Theory and Applications: (Handbook of Statistics 35), Elsevier publications, 2016
3. Judith Hurwitz, Marcia Kaufman, Adrian Bowles, Cognitive Computing and Big Data
4. Analytics, Wiley Publications, 2015
5. Robert A. Wilson, Frank C. Keil, "The MIT Encyclopedia of the Cognitive Sciences",The MIT Press, 1999.
6. Jose Luis Bermúdez, Cognitive Science -An Introduction to the Science of the Mind,
7. Cambridge University Press 2020.

REFERENCES:

1. Noah D. Goodman, Andreas Stuhlmuller, "The Design and Implementation of Probabilistic Programming Languages", Electronic version of book, https://dippl.org/
2. Noah D. Goodman, Joshua B. Tenenbaum, The ProbMods Contributors, "Probabilistic
3. Models of Cognition", Second Edition, 2016, https://probmods.org/.

Course Code	EMOTIONAL INTELLIGENCE	L	T	P	C
CSOE3005		3	0	0	3

COURSE OBJECTIVES:

1. To know the theoretical background of cognition.
2. To understand the link between cognition and computational intelligence.
3. To explore probabilistic programming language.
4. To study the computational inference models of cognition.
5. To study the computational learning models of cognition.

UNIT I INTRODUCTION TO EMOTIONAL INTELLIGENCE   9

Meaning of Emotions, Emotional Intelligence- Importance- Models of Emotional Intelligence- Social Intelligence- IQ and EQ- Self Awareness- Social Skills – Relationship Management- EI and Motivation.

UNIT II UNDERSTANDING EMOTIONS   9

The Brain and Emotion - The Relationship of Mood and Emotion - The Role of Emotion in Organizational Health and the Bottom Line - Types of Emotions- Control of Emotions Gender Differences in Emotion - Impulse Control- Marshmallow Experiment- Negative and Positive Emotions – Emotion and Health

UNIT III MANAGING EMOTIONS   9

Learning EI – Emotional Self Awareness – EI Assessment Tools - Emotional Intelligence and Psychological Adjustment - Issues in Anxiety, Stress, Depression, Anger, Self Esteem and Self Management Empathy

UNIT IV EI PRACTICE IN ORGANIZATIONS   9

Emotional Intelligence and Decision Making - EI and Personality- Work Frustrations- EI and Work Performance- EI and Leadership - EI and Job Stress – EI and Information Processing - EI and Communication – Goal Conflict – EI and Conflict Resolution – EI and Work Place Diversity – Group EI – Star Performers

UNIT V EMOTIONAL COMPETENCE   9

Developing EI in Organization – Transformation and Change – Training, Transfer, Maintenance and Evaluating Change - Emotional Quality Management.

NUMBER OF THEORY PERIODS :   45

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Understand the underlying theory behind cognition.

CO2   Connect to the cognition elements computationally.

CO3   Implement mathematical functions through WebPPL.

CO4   Develop applications using cognitive inference model.

CO5   Develop applications using cognitive learning model.

TEXTB CUM REFERENCE BOOKS:

1. Ciarruchi, J., Forgas, J. and Mayer, John. (2001) Emotional Intelligence in Everyday Life: A Scientific Inquiry. Psychology Press: Philadelphia, PA.

2. Daliph Singh (2001) Emotional Intelligence At Work: A Professional Guide. Response Books: New Delhi

3. Daniel Goleman , (1996) Emotonal Intelligence: Why It can Matter More Than IQ. Bantam Books: NewYork.

4. Doty, G. (2001). Fostering Emotional Intelligence in K-8 Students. Corwin Press: Thousand Oaks, CA.

5. Oatley, K and Jenkins, J (2000) Understanding Emotions. Malden MA: Blackwell

Course Code	DESIGN THINKING	L	T	P	C
CSOE3006		3	0	0	3

COURSE OBJECTIVES:

1. Learn design thinking concepts and principles
2. Use design thinking methods in every stage of the problem
3. Learn the different phases of design thinking
4. Apply various methods in design thinking to different problems

UNIT I INTRODUCTION 9

Why Design? - Four Questions, Ten Tools - Principles of Design Thinking - The process of Design Thinking - How to plan a Design Thinking project.

UNIT II: UNDERSTAND, OBSERVE, AND DEFINE THE PROBLEM 9

Search field determination - Problem clarification - Understanding of the problem - Problem analysis - Reformulation of the problem - Observation Phase - Empathetic design - Tips for observing - Methods for Empathetic Design - Point-of-View Phase - Characterization of the target group - Description of customer needs.

UNIT III IDEATION AND PROTOTYPING 9

Ideate Phase - The creative process and creative principles - Creativity techniques - Evaluation of ideas - Prototype Phase - Lean Startup Method for Prototype Development - Visualization and presentation techniques.

UNIT IV TESTING AND IMPLEMENTATION 9

Test Phase - Tips for interviews - Tips for surveys - Kano Model - Desirability Testing - How to conduct workshops - Requirements for the space - Material requirements - Agility for Design Thinking.

UNIT V FUTURE 9

Design Thinking meets the corporation – The New Social Contract – Design Activism – Designing tomorrow.

NUMBER OF THEORY PERIODS : 45

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1 Grasp the fundamentals of design thinking, including its principles, tools, and project planning.

CO2 Develop skills in problem understanding, observation, analysis, and empathetic design for effective problem-solving.

CO3 Cultivate creativity, ideation, and prototyping capabilities using various techniques and methodologies.

CO4   Acquire proficiency in user testing, workshop facilitation, and agile implementation for successful design execution.

CO5   Explore the future implications of design thinking, including its intersection with corporate innovation, societal responsibilities, and proactive design activism.

TEXTBOOKS:

1. 1. Christian Mueller-Roterberg, Handbook of Design Thinking - Tips & Tools for how to design thinking. [Unit 1, 2, 3, 4]
2. Designing for Growth: a design thinking tool kit for managers By Jeanne Liedtka and Tim Ogilvie. [Unit 1]
3. Change by Design: How Design Thinking Transforms Organizations and Inspires Innovation by Tim Brown. [Unit 5]

REFERENCES:

1. Johnny Schneider, "Understanding Design Thinking, Lean and Agile", O'Reilly Media, 2017.
2. Roger Martin, "The Design of Business: Why Design Thinking is the Next Competitive
3. Advantage", Harvard Business Press , 2009.
4. Hasso Plattner, Christoph Meinel and Larry Leifer (eds), "Design Thinking: Understand – Improve – Apply", Springer, 2011
5. http://ajjuliani.com/design-thinking-activities/
6. 5. https://venturewell.org/class-exercises

Course Code	ELECTRONIC WASTE MANAGEMENT -	L	T	P	C
CSOE3007	ISSUES AND CHALLENGES	3	0	0	3

COURSE OBJECTIVES:

1. To define e-waste, assess its prevalence domestically and internationally, and understand its implications for the environment and public health.
2. To comprehend the legal framework governing e-waste management in India and the responsibilities assigned to different stakeholders.
3. To examine historical and contemporary e-waste disposal, recycling, and recovery methods, including their ecological impact and life cycle assessment.
4. To explore innovative technologies and guidelines to promote environmentally sustainable e-waste management practices.
5. To apply theoretical knowledge through case studies and field exposure, fostering practical skills in devising effective e-waste management strategies.

UNIT I INTRODUCTION 9

What is E-Waste, Indian and global scenario of e-Waste, Growth of Electrical and Electronics industry in India, E-waste generation in India, Composition of e-waste, Possible hazardous substances present in ewaste, Environmental and Health implications

UNIT II E-WASTE LEGISLATION 9

Regulatory regime for e-waste in India, The hazardous waste (Management and Handling) rules 2003, E-waste management rules 2015, Regulatory compliance including roles and responsibility of different stakeholders – producer, manufacturer, consumer etc., Proposed reduction in the use of hazardous substances(RoHS), Extended producer responsibility (EPR).

UNIT III END OF LIFE MANAGEMENT OF E-WASTE 9

Historic methods of waste disposal – dumping, burning, landfill; Recycling and recovery technologies – sorting, crushing, separation; Life cycle assessment of a product – introduction; Case study – optimal planning for computer waste.

UNIT IV ENVIRONMENTALLY SOUND E-WASTE MANAGEMENT 9

Emerging recycling and recovery technologies, Guidelines for environmentally sound management of e-waste, environmentally sound treatment technology for e-waste.

UNIT V APPLICATION CASE STUDY 9

Guidelines for establishment of integrated e-waste recycling and treatment facility, Case studies and unique initiatives from around the world., Case Study and Field Exposure.

NUMBER OF THEORY PERIODS : 45

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1 Understand the origins, composition, and environmental impact of e-waste on both local and global scales.

CO2	Analyse the legislative framework governing e-waste management in India and assess the responsibilities of various stakeholders.
CO3	Evaluate historical and modern methods of e-waste disposal, recycling, and recovery, while considering their ecological footprint.
CO4	Explore emerging technologies and guidelines for environmentally friendly e-waste management, including case studies and global initiatives.
CO5	Apply theoretical knowledge to practical scenarios through case studies and field exposure, developing optimal strategies for managing e-waste.

TEXTBOOKS:

1. Johri R., "E-waste: implications, regulations, and management in India and current global best practices", TERI Press, New Delhi.
2. Choudhary, C. K., Waste management and bioremediation, Oxford Book Company, 2012.

REFERENCES:

1. Wong, J. W. C, Surampalli, R. Y., Zhang, T. C., Tyagi, R. D., Selvan, A., Sustainable Solid Waste Management, American Society of Civil Engineers, 2016.
2. Ojovan, M. I., Handbook of Advanced Radioactive Waste Conditioning Technologies, Woodhead Publishing, 2011.

ONLINE REFERENCES:

1. Dubey, B. K., Electronic Waste Management - Issues and Challenges, Division of Environmental Engineering and Management at Indian Institute of Technology, Kharagpur, NPTEL, https://onlinecourses.nptel.ac.in/noc18_ce07/preview
2. Dubey, B. K., Integrated Waste Management for a Smart City, Division of Environmental Engineering and Management at Indian Institute of Technology, Kharagpur, NPTEL, https://onlinecourses.nptel.ac.in/noc17_ce20/preview
3. Ramachandra, T. V., Municipal Solid Waste Management, Centre for Ecological Sciences, Indian Institute of Science, NPTEL, http://nptel.ac.in/courses/120108005/2#

Course Code	FUNDAMENTALS OF GIS	L	T	P	C
CSOE3008		3	0	0	3

COURSE OBJECTIVES:

1. To define e-waste, assess its prevalence domestically and internationally, and und implications for the environment and public health.
2. To comprehend the legal framework governing e-waste management in Ind responsibilities assigned to different stakeholders.
3. To examine historical and contemporary e-waste disposal, recycling, and recover including their ecological impact and life cycle assessment.
4. To explore innovative technologies and guidelines to promote environmentally sustaina management practices.
5. To apply theoretical knowledge through case studies and field exposure, fostering pract devising effective e-waste management strategies.

UNIT I FUNDAMENTALS OF GIS   **9**

Introduction to GIS - Basic spatial concepts - Coordinate Systems - GIS and Information Systems – Definitions – History of GIS - Components of a GIS – Hardware, Software, Data, People, Methods – Proprietary and open source Software - Types of data – Spatial, Attribute data- types of attributes – scales/ levels of measurements.

UNIT II SPATIAL DATA MODELS   **9**

Database Structures – Relational, Object Oriented – ER diagram - spatial data models – Raster Data Structures – Raster Data Compression - Vector Data Structures - Raster vs Vector Models- TIN and GRID data models - OGC standards - Data Quality.

UNIT III DATA INPUT AND TOPOLOGY   **9**

Scanner - Raster Data Input – Raster Data File Formats – Vector Data Input –Digitiser – Topology - Adjacency, connectivity and containment – Topological Consistency rules – Attribute Data linking – ODBC – GPS - Concept GPS based mapping.

UNIT IV DATA ANALYSIS   **9**

Vector Data Analysis tools - Data Analysis tools - Network Analysis - Digital Education models - 3D data collection and utilisation.

UNIT V APPLICATIONS   **9**

GIS Applicant - Natural Resource Management - Engineering - Navigation - Vehicle tracking and fleet management - Marketing and Business applications - Case studies.

NUMBER OF THEORY PERIODS : 45

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Have basic idea about the fundamentals of GIS.
CO2   Understand the types of data models.
CO3   Get knowledge about data input and topology.
CO4   Gain knowledge on data quality and standards.
CO5   Understand data management functions and data output

TEXTBOOKS:

1. Kang - Tsung Chang, Introduction to Geographic Information Systems, McGraw Hill Publishing, 2nd Edition, 2011.
2. Ian Heywood, Sarah Cornelius, Steve Carver, Srinivasa Raju, "An Introduction Geographical Information Systems, Pearson Education, 2nd Edition,2007.

REFERENCES:

1. Lo.C.P., Albert K.W. Yeung, Concepts and Techniques of Geographic Information Systems, Prentice-Hall India Publishers, 2006

Course Code	BIOINFORMATICS	L	T	P	C
CSOE3009		3	0	0	3

COURSE OBJECTIVES:

1. Understand the fundamental concepts of bioinformatics and computational biology, including biological sequences and the classification of biological databases.
2. Master various sequence analysis techniques such as sequence alignment, substitution matrices, and database similarity searching tools for interpreting biological data effectively.
3. Explore molecular phylogeny principles and methods, including distance matrix methods and character-based approaches, for constructing evolutionary relationships among biological sequences.
4. Develop proficiency in analyzing macromolecular structures, including gene prediction, conserved domain analysis, and protein structure prediction techniques, along with validating predicted structures.
5. Apply bioinformatics approaches to diverse biological applications such as systems biology, microarray data analysis, DNA computing, and drug discovery, showcasing their significance in advancing biological research and technology.

UNIT I BIOLOGICAL DATABASES   9

Introduction to Bioinformatics and Computational Biology, Biological sequences, Classification of biological databases - Sequence Databases, Structure Databases, Genome specific databases, Special Databases and applications- Microarray, Metabolic pathway, motif, and domain databases, Data file formats.

UNIT II SEQUENCE ANALYSIS   9

Sequence Alignment- Homology vs Similarity, Similarity vs Identity. Types of Sequence alignment - Pairwise and Multiple sequence alignment, Global alignment, Local alignment, Dotplot, Alignment algorithms- Needleman Wunsch and Smith and waterman algorithm, Substitution matrices- PAM, BLOSUM. Multiple Sequence Alignment- Application of multiple alignments, Viewing and editing of MSA and Scoring function. Database Similarity Searching-Basic Local Alignment Search Tool (BLAST), FASTA, PHI BLAST, PSI BLAST, BLAST algorithm.

UNIT III MOLECULAR PHYLOGENY   9

Phylogenetics Basics, Molecular clock theory, Ultrametric trees, Distance matrix methods UPGMA, NJ, Character based methods-Maximum Parsimony. Methods of evaluating phylogenetic methods- boot strapping, jack-knifing.

UNIT IV MACROMOLECULAR STRUCTURE ANALYSIS   9

Gene prediction, Conserved domain analysis, Protein structure visualization, Prediction of protein secondary structure, Tertiary structure prediction- Homology modelling, Threading, Ab-initio prediction. Validation of the predicted structure using Ramachandran plot, stereochemical properties, Structure- structure alignment.

UNIT V APPLICATIONS   9

Introduction to Systems Biology and Synthetic Biology, Microarray data analysis, DNA

computing, Bioinformatics approaches for drug discovery, Applications of Bioinformatics in genomics and proteomics- Assembling the genome, STS content mapping for clone contigs, Functional annotation, Peptide mass fingerprinting.

NUMBER OF THEORY PERIODS : 45

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Apply computational based solutions for biological perspectives

CO2   Analyze the evolutionary relationship between the organisms

CO3   Understand the macromolecules structure prediction methods

CO4   Relate how bioinformatics methods can be used in sequence to structure and function analysis.

CO5   Learn the applications of bioinformatics approach for drug discovery, genomics and proteomics.

TEXT CUM REFERENCE BOOKS:

1. Arthur K. Lesk- Introduction to Bioinformatics, Oxford University Press.

2. Baxivanis and FouletteD- Bioinformatics: A Practical Guide to the Analysis of Genes and Proteins, Wiely Indian Edition, 2001.

3. Attwood TK, parry DJ-Smith, Introduction to Bioinformatics, Pearson Education, 1st Edition, 11th Reprint 2005.

4. 4. David W. Mount-Bioinformatics Sequence and Genome Analysis, Cold Spring Harbor Laboratory Press. 1st edition 2001.

Open Elective II

Course Code	BUSINESS STRATEGY	L	T	P	C
CSOE3010		3	0	0	3

COURSE OBJECTIVES:

1. To introduce the concepts of strategic management and understand its nature in competitive and institutional landscape.
2. To develop a holistic approach to see business issues comprehensively and using other core and functional subject knowledge for decision-making.
3. To identify and interpret the critical challenges and opportunities before an organization.

UNIT I INTRODUCTION TO STRATEGIC MANAGEMENT   9

Importance of Strategic Management, Vision and Objectives, Schools of thought in Strategic Management, strategy Content, Process, and Practice, Fit Concept and Configuration Perspective in Strategic Management

UNIT II INTERNAL ENVIRONMENT OF FIRM- RECOGNIZING A FIRM'S INTELLECTUAL   9 ASSETS

Core Competence as the Root of Competitive Advantage, Sources of Sustained Competitive Advantage, Business Processes and Capabilities-based Approach to Strategy

UNIT III EXTERNAL ENVIRONMENTS OF FIRM- COMPETITIVE STRATEGY   9

Five Forces of Industry Attractiveness that Shape Strategy, The concept of Strategic Groups, and Industry Life Cycle, Generic Strategies, Generic Strategies and the Value Chain.

UNIT IV CORPORATE STRATEGY, AND GROWTH STRATEGIES   9

The Motive for Diversification, Related and Unrelated Diversification, Business Portfolio Analysis, Expansion, Integration and Diversification, Strategic Alliances, Joint Ventures, and Mergers & Acquisitions.

UNIT V STRATEGY IMPLEMENTATION: STRUCTURE AND SYSTEMS   9

The 7S Framework, Strategic Control and Corporate Governance.

NUMBER OF THEORY PERIODS: 45

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1 Learn the fundamental concepts of strategic management to analyze business situations and apply these concepts to solve business problems.

CO2 Understand the fundamental principles of and interrelationships among business functions such as:R&D, production, marketing, finance, HR and information technology

CO3 Understand the inter-relationships of business to individuals, other organizations,

government and society.

CO4   Describe the tools of strategic analysis thoroughly, how they are used, and where they fit in the managerial process to frame and implement strategies.

TEXTBOOKS:

1. Strategic management of technological innovation (2019), Schilling, M. A., & Shankar, R,McGraw-Hill Education.
2. The business of platforms: Strategy in the age of digital competition, innovation, and power (2019), Cusumano, M. A., Gawer, A., & Yoffie, D. B.,New York: Harper Business.

REFERENCES:

1. Dislodging multinationals: India's strategy in comparative perspective (2019), Encarnation, D.Cornell, University Press.
2. Dynamics of knowledge intensive entrepreneurship: Business strategy and public policy (2018),
3. Malerba, F., Caloghirou, Y., McKelvey, M., & Radoševic, S. (Eds.), Routledge.

Course Code	INTRODUCTION TO CYBERSECURITY	L	T	P	C
CSOE3011		3	0	2	4

COURSE OBJECTIVES:

1. To develop a comprehensive understanding of fundamental cybersecurity concepts, including cyber threats, preventive measures, and cyber security principles.

2. To explore networking basics, focusing on concepts, protocols, and architectures crucial to understanding cyber security measures.

3. To acquire knowledge on core security principles, including risk management practices, to build a solid foundation for implementing effective security measures.

4. To understand the principles of cryptography and encryption, exploring their role in securing data and communication channels in cybersecurity.

5. To develop the skills to create, implement, and enforce security policies and procedures, fostering a security-aware culture and mitigating cyber threats through effective governance.

UNIT I INTRODUCTION TO CYBERSECURITY CONCEPTS 9

Introduction to Cybersecurity Fundamentals - Cyber Threat Landscape -Cybersecurity Frameworks and Standards - Security Architecture and Models - Incident Response and Cybersecurity Incident Handling - Security Awareness and Training - Legal and Ethical Aspects of Cybersecurity - Emerging Trends in Cybersecurity.

UNIT II FUNDAMENTALS OF NETWORKING 9

Introduction to Networking Concepts - OSI Model Overview - TCP/IP Protocol Suite -

Network Devices and Components - IP Addressing and Subnetting - Routing and Switching Basics - Wireless Networking Fundamentals - Network Security Principles.

UNIT III SECURITY FUNDAMENTALS 9

Introduction to Security Principles - Access Control and Authentication - Security Risk Management - Security Policies and Procedures - Security Incident Response - Security Awareness Training - Vulnerability Assessment and Management - Physical Security Considerations.

UNIT IV CRYPTOGRAPHY AND ENCRYPTION 9

Introduction to cryptography - symmetric encryption algorithms - asymmetric encryption and public key infrastructure (PKI) - hash functions and message digests - digital signatures

- cryptographic key management - transport layer security (TLS) and secure sockets layer (SSL) - cryptography in blockchain technology.

UNIT V SECURITY POLICIES AND PROCEDURES 9

Introduction to security policies and procedures - policy development and implementation - access control policies - data classification and handling policies - incident response plans - security awareness training programs - compliance and regulatory policies - security auditing and monitoring procedures.

NUMBER OF THEORY PERIODS :   45

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Gain a clear understanding of fundamental cybersecurity principles, exploring the core concepts that form the basis of cyber defense

CO2   Understand the application of cybersecurity policies and risk management practices, emphasizing their importance in maintaining a secure digital environment

CO3   Learn about the basic tools and techniques for investigating various cybercrimes, providing a foundational knowledge base for cybercrime detection and resolution

CO4   Apply security fundamentals to digital environments, incorporating knowledge of

risk management, access control, and other core principles to enhance overall security

CO5   Demonstrate proficiency in cyber forensics techniques and methodologies, integrating knowledge from security policies and procedures to enhance investigative capabilities

TEXTBOOKS LINKS:

Unit 1: Introduction to Cybersecurity by jeetendra pande
https://uou.ac.in/sites/default/files/slm/Introduction-cyber-security.pdf
Unit 2: Fundamentals of Networking:
https://www.cisco.com/c/dam/global/fi_fi/assets/docs/SMB_University_120307_Networking_Fundamentals.pdf
Unit 3: Security Fundamentals:
https://training.apnic.net/wp-content/uploads/sites/2/2016/11/eSEC01_NetSec.pdf Unit
4: **Cryptography and Encryption:**
https://www.cs.umd.edu/~waa/414-F11/IntroToCrypto.pdf
Unit 5: **Security Policies and Procedures:**
https://oklahoma.gov/content/dam/ok/en/omes/documents/InfoSecPPG.pdf

REFERENCES:

1. Diogenes, Y., & Ozkaya, E. (2018). Cybersecurity - Attack and defense strategies: Infrastructure security with red team and blue team tactics. Packt Publishing.
2. Schneier, B. (2017). Applied cryptography: Protocols, algorithms and source code in C. John Wiley & Sons.
3. Erickson, J. (2008). Hacking: The art of exploitation (2nd ed.). No Starch Press.

Course Code	GREEN COMPUTING	L	T	P	C
CSOE3012		3	0	0	3

COURSE OBJECTIVES:

1. Understand the fundamentals, significance, and regulations shaping the field of green computing.
2. Implement techniques to enhance energy efficiency in computing systems and embrace sustainable practices.
3. Develop eco-friendly software solutions, incorporating energy-efficient coding practices and sustainable software design.
4. Design and manage data centers with a focus on energy efficiency and environmentally responsible practices.
5. Equip students with knowledge to manage electronic waste responsibly, emphasizing reuse, recycling, and regulatory compliance.

UNIT I FOUNDATIONS OF GREEN COMPUTING   **9**

Introduction to Green Computing, Significance and Global Relevance, Environmental Challenges in Computing, Energy Consumption in Computing, Electronic Waste and Resource Depletion, Green Policies and Regulations, Compliance with Environmental Standards, Case Studies in Sustainable Computing.

UNIT II OPTIMIZING ENERGY CONSUMPTION   **9**

Techniques for Reducing Energy Consumption, Power Management Strategies, Energy-efficient Hardware Design, Sustainable Software Development, Virtualization and Server Consolidation, Energy-aware Algorithms, Renewable Energy Integration, Best Practices in Energy Optimization.

UNIT III GREEN SOFTWARE PRACTICES   **9**

Principles of Green Software, Code Optimization for Energy Efficiency, Sustainable Software Design Patterns, Energy-efficient Algorithms, Software Development Lifecycle and Green Computing, Case Studies in Green Software Development, Hands-on Coding Exercise, Collaboration Between Developers and Environmental Experts

UNIT IV BUILDING ECO-FRIENDLY DATA CENTERS   **9**

Green Data Center Concepts, Cooling and Climate Control Innovations, Server Virtualization and Consolidation, Sustainable Server and Storage Solutions, Energy-efficient Hardware Selection, Monitoring and Optimization Tools, Data Center Certification Standards, Group Workshop: Designing a Green Data Center.

UNIT V RESPONSIBLE E-WASTE PRACTICES   **9**

Understanding Electronic Waste (E-waste), E-waste Collection and Recycling Programs, Reuse and Upcycling of Computing Equipment, Eco-friendly Disposal Practices, Regulatory Compliance in E-waste Management, Community Engagement for E-waste Awareness, Student-led E-waste Collection Drive, Final Project Presentations and Reflections.

NUMBER OF THEORY PERIODS :   45

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Understand environmental impact, regulations, and ethical considerations influencing sustainable computing practices globally.

CO2   Apply strategies for energy efficiency and renewable energy integration in computing systems effectively.

CO3   Develop eco-friendly software, employing energy-efficient coding, sustainable design patterns, and lifecycle assessment principles.

CO4   Design and manage data centers with a focus on energy efficiency, eco-friendly hardware, and certifications.

CO5   Lead e-waste initiatives, ensuring ethical disposal, regulatory compliance, and community engagement for sustainability.

TEXTBOOKS:

1. "Green IT for Dummies" by Carol Baroudi (2022)
2. "Energy Efficient Servers: Blueprints for Data Center Optimization" by Corey Gough (2021)
3. "Sustainable Software Development: An Agile Perspective" by Kevin Carlson (2023)
4. "Designing Green Data Centers" by Bill Kleyman (2022)
5. "E-Waste in Transition: From Pollution to Resource" by Flor Avelino (2021)

REFERENCES:

1. "Green Computing: Tools and Techniques for Saving Energy, Money, and Resources" by Bud E. Smith (2023)
2. "Energy-Efficient Distributed Computing Systems" by Albert Y. Zomaya (2022)
3. "Sustainable Software Architecture: A Framework for Modern Enterprise Applications" by Carola Lilienthal (2023)
4. "Data Center Handbook" by Hwaiyu Geng (2021)
5. "Electronic Waste Management and Treatment Technology" by Majeti Narasimha Vara Prasad (2022)

Course Code	E – BUSINESS MANAGEMENT	L	T	P	C
CSOE3013		3	0	0	3

COURSE OBJECTIVES:

1. Define the concepts of e-business and e-commerce, analyze economic factors and advantages, and debunk myths surrounding e-business models.
2. Explore the technology infrastructure supporting e-business, including internet protocols, intranets, extranets, and web server hardware/software.
3. Evaluate consumer-oriented and business-oriented e-business applications such as e-tailing, e-CRM, and e-governance, along with strategies for marketing and delivery management.
4. Examine e-business payment systems and security measures, including characteristics, protocols, and internet security techniques like cryptography and network security.
5. Analyze legal, ethical, and privacy issues related to e-business, including consumer protection, cyber laws, taxation policies, and encryption regulations.

UNIT I INTRODUCTION TO e-BUSINESS   9

e-business, e-business vs e-commerce, Economic forces – advantages – myths – e-business models, design, develop and manage e-business, Web 2.0 and Social Networking, Mobile Commerce, S-commerce.

UNIT II TECHNOLOGY INFRASTRUCTURE   9

Internet and World Wide Web, internet protocols - FTP, intranet and extranet, information publishing technology- basics of web server hardware and software.

UNIT III BUSINESS APPLICATIONS   9

Consumer oriented e-business – e-tailing and models - Marketing on web – advertising, e-mail marketing, affiliated programs - e-CRM; online services, Business oriented e-business, e-governance, EDI on the internet, Delivery management system, Web Auctions, Virtual communities and Web portals – social media marketing

UNIT IV e-BUSINESS PAYMENTS AND SECURITY   9

E-payments - Characteristics of payment of systems, protocols, e-cash, e-cheque and Micro payment systems- internet security – cryptography – security protocols – network security.

UNIT V LEGAL AND PRIVACY ISSUES   9

Legal, Ethics and privacy issues – Protection needs and methodology – consumer protection, cyber laws, contracts and warranties, Taxation and encryption policies.

NUMBER OF THEORY PERIODS : 45

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Ability to build and manage an e-business.

CO2   Knowledge about Technology Infrastructure
CO3   Understanding of customer-oriented business applications
CO4   Knowledge of e business payment protocols and security
CO5   Understanding of ethical, legal , privacy issues and encryption policies

TEXTBOOKS:

1. Harvey M.Deitel, Paul J.Deitel, Kate Steinbuhler, e-business and e-commerce for managers, Pearson, 2011.

2. Efraim Turban, Jae K. Lee, David King, Ting Peng Liang, Deborrah Turban, Electronic Commerce –A managerial perspective, Pearson Education Asia, 2010.

REFERENCES:

1. Parag Kulkarni, SunitaJahirabadkao, Pradeep Chande, e business, Oxford University Press, 2012.
2. Hentry Chan &el , E-Commerce – fundamentals and Applications, Wiley India Pvt Ltd, 2007.
3. Gary P. Schneider, Electronic commerce, Thomson course technology, Fourth annual edition, 2007
4. Bharat Bhasker, Electronic Commerce – Frame work technologies and Applications, 3rd Edition. Tata McGrawHill Publications, 2009
5. KamleshK.Bajaj and Debjani Nag, Ecommerce- the cutting edge of Business, Tata McGraw Hill Publications, 7th reprint, 2009.
6. Kalakota et al, Frontiers of Electronic Commerce, Addison Wesley, 2004
7. MichealPapaloelon and Peter Robert, e-business, Wiley India, 2006.

Course Code	MULTICORE ARCHITECTURE AND PROGRAMMING	L	T	P	C
CSOE3014		3	0	0	3

COURSE OBJECTIVES:

1. To understand the need for multi-core processors, and their architecture.
2. To understand the challenges in parallel and multithreaded programming.
3. To learn about the various parallel programming paradigms,
4. To develop multicore programs and design parallel solutions.

UNIT I MULTI-CORE PROCESSORS **9**

Single core to Multi-core architectures – SIMD and MIMD systems – Interconnection networks – Symmetric and Distributed Shared Memory Architectures – Cache coherence – Performance

Issues – Parallel program design.

9

UNIT II PARALLEL PROGRAM CHALLENGES

Performance – Scalability – Synchronization and data sharing – Data races – Synchronization

primitives (mutexes, locks, semaphores, barriers) – deadlocks and livelocks – communication

between threads (condition variables, signals, message queues and pipes).

UNIT III SHARED MEMORY PROGRAMMING WITH OPENMP **9**

OpenMP Execution Model – Memory Model – OpenMP Directives – Work-sharing Constructs – Library functions – Handling Data and Functional Parallelism – Handling Loops – Performance Considerations.

UNIT IV DISTRIBUTED MEMORY PROGRAMMING WITH MPI **9**

MPI program execution – MPI constructs – libraries – MPI send and receive – Point-to-point and Collective communication – MPI derived datatypes – Performance evaluation.

UNIT V PARALLEL PROGRAM DEVELOPMENT **9**

Case studies – n-Body solvers – Tree Search – OpenMP and MPI implementations and comparison.

NUMBER OF THEORY PERIODS : 45

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Describe multicore architectures and identify their characteristics and challenges.

CO2   Identify the issues in programming Parallel Processors.

CO3   Write programs using OpenMP and MPI.

CO4   Design parallel programming solutions to common problems.

CO5   Compare and contrast programming for serial processors and programming for parallel processors.

TEXT CUM REFERENCE BOOKS:

1. Peter S. Pacheco, "An Introduction to Parallel Programming, Morgan-Kauffman/Elsevier, 2021.
2. Darryl Gove, "Multicore Application Programming for Windows, Linux, and Oracle Solaris, Pearson, 2011 (unit 2)
3. Michael J Quinn, "Parallel programming in C with MPI and OpenMP, Tata McGraw Hill,2003. Tentative
4. Victor Alessandrini, Shared Memory Application Programming, 1st Edition, Concepts and Strategies in Multicore Application Programming, Morgan Kaufmann, 2015.
5. Yan Solihin, Fundamentals of Parallel Multicore Architecture, CRC Press, 2015

Course Code	**AUGMENTED REALITY AND VIRTUAL REALITY**	**L**	**T**	**P**	**C**
CSOE3015		3	0	0	3

COURSE OBJECTIVES:

1. Define AR and VR, explore historical evolution, and understand their significance and applications.
2. Master AR and VR technologies, utilizing development tools and frameworks effectively.
3. Apply UI/UX design principles, creating immersive narratives and interactive experiences in AR and VR.
4. Evaluate AR/VR applications in healthcare, education, and industries, understanding practical use cases.
5. Anticipate emerging technologies, propose solutions to challenges, and assess the future impact of AR and VR.

UNIT I INTRODUCTION TO AR AND VR TECHNOLOGIES 9

Definitions and Fundamentals of AR and VR, Historical Evolution of AR and VR, Significance and Real-world Applications, Social Impact and Ethical Considerations, Key Technological Components, Overview of AR and VR Devices, Comparison of AR and VR Experiences, Future Trends in AR and VR.

UNIT II CORE TECHNOLOGIES AND DEVELOPMENT TOOLS 9

Augmented Reality (AR) Technologies: Marker-based and Markerless, Virtual Reality (VR) Technologies: Headsets and Controllers, AR and VR Development Tools and Frameworks, Cross-cutting Technologies: AI and Machine Learning in AR/VR, Advances in Spatial Computing, 3D Modeling and Content Creation, Building AR and VR Environments, Hands-on Development Exercise

UNIT III DESIGN PRINCIPLES FOR AR AND VR EXPERIENCES 9

User Interface (UI) Design Principles, User Experience (UX) Design Principles, Interaction Design in AR and VR, Gestures, Voice, and Haptic Feedback, Storytelling in Immersive Environments, Content Creation for AR and VR, Prototyping AR and VR Experiences, Evaluation of AR and VR Designs

UNIT IV APPLICATIONS IN INDUSTRY AND EDUCATION 9

AR/VR in Healthcare: Training and Patient Care, AR/VR in Education and Training, Virtual Conferencing and Collaboration, Industrial Applications of AR and VR, Retail and Marketing in AR and VR, Architectural Visualization, Gaming and Entertainment, Industry-specific Use Cases and Case Studies

UNIT V FUTURE TRENDS AND CHALLENGES IN AR AND VR 9

Emerging Technologies: AR Glasses, Smart Contact Lenses, Challenges and Solutions in AR and VR Development, Scalability Issues and Solutions, Ethical Considerations in AR and VR, Societal Impact of Widespread Adoption, The Integration of AR and VR with Other Technologies, Future of Work and Education in AR and VR, Student-led Discussion and Project Exploration.

NUMBER OF THEORY PERIODS : 45

SAMPLE LIST OF EXPERIMENTS:

ASSIGNMENTS / PROJECTS:

Assignment 1: Fundamentals Review (3 hours)
Review foundational concepts in AR and VR through quizzes and short answer questions.
Hands-on Project 1: AR Basics (6 hours)
Develop a basic AR application using marker-based technology with guidance and feedback.

Assignment 2: VR Development Principles (3 hours)
Analyze VR development principles and submit a reflective assignment on key concepts.
Hands-on Project 2: VR Environment Creation (6 hours)
Design and build a virtual environment, integrating UI/UX principles and interactive elements.

Assignment 3: Design Critique (3 hours)
Critique a given AR/VR design, highlighting strengths, weaknesses, and potential improvements.
Hands-on Project 3: Industry-specific Application (6 hours)
Develop a practical AR or VR application targeting a specific industry, emphasizing real-world relevance.

Assignment 4: Future Trends Paper (3 hours)
Research and write a paper on emerging trends in AR and VR, addressing their future impact.

COURSE OUTCOMES:

On completion of this course, the students will be able to:
CO1   Establish a foundational knowledge of AR and VR, including key terms, historical context, and ethical considerations.
CO2   Demonstrate proficiency in using AR and VR technologies, development tools, and frameworks in practical applications
CO3   Apply advanced UI/UX design principles, developing compelling narratives and interactive elements for immersive AR and VR experiences
CO4   Analyze and compare industry-specific applications of AR and VR, presenting insights into their impact and potential improvements.

CO5   Evaluate evolving technologies, propose innovative solutions to challenges, and predict societal implications of advancing AR and VR.

TEXTBOOKS:

1. "Augmented Reality: Principles and Practice" - Dieter Schmalstieg and Tobias Hollerer (2021)
2. "Virtual Reality: Concepts and Technologies" - William R. Sherman and Alan B. Craig (2022)
3. "Designing Virtual Worlds" - Richard Bartle (2018)
4. "The Fourth Transformation: How Augmented Reality & Artificial Intelligence Will Change Everything" - Robert Scoble and Shel Israel (2016)
5. "Blockchain Basics: A Non-Technical Introduction in 25 Steps" - Daniel Drescher (2020)

REFERENCES:

1. "Understanding Augmented Reality: Concepts and Applications" - Alan B. Craig (2013)
2. "Unity Virtual Reality Projects" - Jonathan Linowes (2015)
3. "Virtual Reality Designer: Leveraging VR for Ongoing Design Innovation" - Todd Richmond (2018)
4. "Augmented Reality in Education: Cases, Places, and Lessons" - Charles Wankel and Ryan Crowley (2014)
5. "The Age of Em: Work, Love, and Life when Robots Rule the Earth" - Robin Hanson (2016)

Course Code	INTERNET OF THINGS (IOT)	L	T	P	C
CSOE3016		3	0	0	3

COURSE OBJECTIVES:

1. To give an understanding about the choice and application of IoT & M2M communication protocols.
2. To describe Cloud computing and design principles of IoT.
3. To relate to MQTT clients, MQTT server and its programming.
4. To describe the architectures and communication protocols of WSNs.
5. To identify the uplink and downlink communication protocols associated with specific application of IOT /WSNs.

UNIT I OVERVIEW OF INTERNET OF THINGS   **9**

IoT Conceptual Framework, IoT Architectural View, Technology Behind IoT, Sources of IoT,M2M communication, Examples of IoT. Modified OSI Model for the IoT/M2M Systems, data enrichment, data consolidation and device management at IoT/M2M Gateway, web communication protocols used by connected IoT/M2M devices, Message communication protocols (CoAP-SMS, CoAP-MQ, MQTT,XMPP) for IoT/M2M devices. -Refer Chapter 1,2 and 3 of Text 1.

UNIT II ARCHITECTURE AND DESIGN PRINCIPLES FOR IOT   **9**

Internet connectivity, Internet based communication, IPv4, IPv6,6LoWPAN protocol, IP Addressing in the IoT, Application layer protocols: HTTP, HTTPS,FTP, TELNET and ports.

UNIT III PROTOTYPING AND DESIGNING SOFTWARE FOR IOT APPLICATIONS   **9**

Introduction, Prototyping Embedded device software, Programming Embedded Device Arduino Platform using IDE, Reading data from sensors and devices, Devices, Gateways, Internet and Web/Cloud services software development. Programming MQTT clients and MQTT server. Introduction to IoT privacy and security. Vulnerabilities, security requirements and threat analysis, IoT Security Tomography and layered attacker model. -Refer Chapter 9 and 10 of Text 1.

UNIT IV OVERVIEW OF WIRELESS SENSOR NETWORKS   **9**

Challenges for Wireless Sensor Networks, Enabling Technologies for Wireless Sensor Networks. Architectures: Single-Node Architecture - Hardware Components, Energy Consumption of Sensor Nodes, Operating Systems and Execution Environments, Network Architecture-Sensor Network Scenarios, Optimization Goals and Figures of Merit, Design principles for WSNs, Service interfaces of WSNs Gateway Concepts. -Refer Chapter 1,2, 3 of Text 2.

UNIT V COMMUNICATION PROTOCOLS   **9**

Physical Layer and Transceiver Design Considerations, MAC Protocols for Wrreless Sensor Networks, Low Duty Cycle Protocols and Wakeup Concepts - S-MAC, The Mediation Device Protocol, Wakeup Radio Concepts, Contention based protocols (CSMA, PAMAS), Schedule based protocols (LEACH, SMACS, TRAMA) Address and Name Management in WSNs, Assignment of MAC Addresses, Routing Protocols- Energy-Efficient Routing, Geographic Routing, Hierarchical networks by clustering. Refer Chapter 4, 5, 7 and 11 of Text 2. Lt, L2, L3

NUMBER OF THEORY PERIODS : 45

SAMPLE LIST OF EXPERIMENTS:

1. Implement an IoT conceptual framework and analyze M2M communication using CoAP-SMS and CoAP-MQ protocols, focusing on data enrichment and consolidation at the IoT/M2M gateway.

2. Design a prototype for Internet connectivity in IoT, exploring IPv4, IPv6, and 6LoWPAN protocol, alongside understanding application layer protocols like HTTP, HTTPS, and FTP.

3. Develop embedded device software using Arduino IDE, read data from sensors, and program MQTT clients and servers while addressing IoT privacy and security concerns through threat analysis and IoT Security Tomography.

4. Explore enabling technologies and architectures for wireless sensor networks (WSNs), understand single-node architecture components, and optimize energy consumption using operating systems and execution environments.

5. Analyze physical layer and transceiver design considerations for WSNs, implement MAC protocols like S-MAC and contention-based protocols like CSMA, and design energy-efficient routing protocols for hierarchical networks by clustering.

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Understand choice and application of IoT & M2M communication protocols.

CO2   Describe Cloud computing and design principles of IoT.

CO3   Relate to MQTT clients, MQTT server and its programming.

CO4   Describe the architectures and communication protocols of WSNs.

CO5   Identify the uplink and downlink communication protocols associated with specific application of IOT /WSNs.

TEXTBOOKS:

1. Raj Kamal, "Internet of Things-Architecture and Design Principles", McGraw Hill Education.
2 Holger Karl & Andreas Willig, " Protocols And Architectures for Wireless Sensor Networks", John Wiley, 2005.

REFERENCES:

1. Feng Zhao & Leonidas J. Guibas, "Wireless Sensor Networks- An Information Processing Approach", Elsevier, 007.
2 Kazem Sohraby, Daniel Minoli, & Taieb Znati, '' Wireless Sensor Networks- Technology, Protocols and Applications", John Wiley, 2007.
3. Anna Hac, 'Wireless Sensor Network Designs", John Wiley, 2003.

Course Code	**CYBER LAW**	L	T	P	C
CSOE3017		3	0	0	3

COURSE OBJECTIVES:

	1.	To gain through knowledge on the basic concepts which lead to the formation and execution of electronic contracts
	2.	To Study the Technical aspects of Cyber Security and Evidence Aspects
	3.	To acquire knowledge on Information Technology Act and EDI

UNIT I INTRODUCTION   **9**

Cyber Law: Introduction- Concept of Cyberspace-E-Commerce in India-Privacy factors in E - Commerce-cyber law in E-Commerce-Contract Aspects.

UNIT II SECURITY ASPECTS   **9**

Security Aspects: Introduction-Technical aspects of Encryption-Digital Signature-Data Security. Intellectual Property Aspects: WIPO-GII-ECMS-Indian Copy rights act on soft propriety worksIndian Patents act on soft propriety works.

UNIT III EVIDENCE ASPECTS   **9**

Evidence Aspects: Evidence as part of the law of procedures –Applicability of the law of Evidence on Electronic Records-The Indian Evidence Act1872.Criminal aspect: Computer Crime-Factors influencing Computer Crime- Strategy for prevention of computer crime Amendments to Indian Penal code 1860.

UNIT IV GLOBAL TRENDS   **9**

Global Trends- Legal frame work for Electronic Data Interchange: EDI Mechanism-Electronic Data Interchange Scenario in India

UNIT V THE INFORMATION TECHNOLOGY ACT 2000   **9**

The Information Technology Act 2000-Definitions-Authentication of Electronic Records Electronic Governance-Digital Signature Certificates. Expert lectures, online seminars – webinars

NUMBER OF THEORY PERIODS :   45

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Discuss the concepts of Cyber law and Cyber Space

CO2   Describe Cyber Security technical aspects.

CO3   Explain the Evidence Aspects.

CO4   Understand the Electronic Data Interchange Scenario in India.

CO5   To gain knowledge on Information Technology Act.

TEXTBOOKS:

1. The Indian Cyber Law : Suresh T.Viswanathan, Bharat Law House, New Delhi.

REFERENCES:

1. https://www.youtube.com/watch?v=KtuCsBlJXk8
2. https://www.youtube.com/watch?v=6srnawS4PLQ&list=PLX0Im12KwTwlmjOWfFqejg8go
   7JBj72J
3. 3 https://www.youtube.com/watch?v=SCgc55vtd6M

APPENDIX C: MANDATORY COURSES

Mandatory Course – I (MC)

S. No.	Course Code	Course Title	Semester	L	T	P	C
1.	MC4301	Introduction to Women and Gender Studies	III	2	0	0	0
2.	MC4302	Elements of Literature	III	2	0	0	0
3.	MC4303	Film Appreciation	III	2	0	0	0
4.	MC4304	Disaster Risk Reduction and Management	III	2	0	0	0
5.	MC4305	Environmental Sciences and Sustainability	III	2	0	0	0
Total Credits				**0**			

Mandatory Course – II (MC)

S. No.	Course Code	Course Title	Semester	L	T	P	C
1.	MC4401	Well Being with Traditional Practices Yoga Ayurveda and Sidda	IV	2	0	0	0
2.	MC4402	History of Science and Technology in India	IV	2	0	0	0
3.	MC4403	Political and Economic Thought for a Human Society	IV	2	0	0	0
4.	MC4404	State, Nation Building and Politics in India	IV	2	0	0	0
5.	MC4405	Industrial Safety	IV	2	0	0	0
Total Credits				**0**			

Course Code	**INTRODUCTION TO WOMEN AND GENDER STUDIES**	L	T	P	C
MC4301		3	0	0	0

UNIT I CONCEPTS

2+4

Sex vs. Gender, masculinity, femininity, socialization, patriarchy, public/ private, essentialism, binaryism, power, hegemony, hierarchy, stereotype, gender roles, gender relation, deconstruction, resistance, sexual division of labour.

UNIT II FEMINIST THEORY

9

Liberal, Marxist, Socialist, Radical, Psychoanalytic, postmodernist, ecofeminist.

UNIT III WOMEN'S MOVEMENTS: GLOBAL, NATIONAL AND LOCAL

4+4

Rise of Feminism in Europe and America.
Women's Movement in India.

UNIT IV GENDER AND LANGUAGE

3+4

Linguistic Forms and Gender.
Gender and narratives.

UNIT V GENDER AND REPRESENTATION

9

Advertising and popular visual media.
Gender and Representation in Alternative Media.
Gender and social media.

TOTAL NUMBER OF PERIODS: **45**

Course Code	**ELEMENTS OF LITERATURE Y**	**L**	**T**	**P**	**C**
MC4302		3	0	0	0

COURSE OBJECTIVES:

1. To make the students aware about the finer sensibilities of human existence through an art form. The students will learn to appreciate different forms of literature as suitable modes of expressing human experience.

UNIT I INTRODUCTION TO ELEMENTS OF LITERATURE   **9**

Relevance of literature a) Enhances Reading, thinking, discussing and writing skills. b) Develops finer sensibility for better human relationship. c) Increases understanding of the problem of humanity without bias. d) Providing space to reconcile and get a cathartic effect.

UNIT II ELEMENTS OF FICTION   **9**

a) Fiction, fact and literary truth.
b) Fictional modes and patterns.
c) Plot character and perspective.

UNIT III ELEMENTS OF POETRY   **9**

a) Emotions and imaginations. b) Figurative language. c) (Simile, metaphor, conceit, symbol, pun and irony). d) Personification and animation. e) Rhetoric and trend.

UNIT IV ELEMENTS OF DRAMA   **9**

a) Drama as representational art. b) Content mode and elements. c) Theatrical performance. d) Drama as narration, mediation and persuasion. e) Features of tragedy, comedy and satire.

UNIT V OTHER SESSION   **9**

Tutorials:
Laboratory:
Project: The students will write a term paper to show their understanding of a particular piece of literature
ASSESSMENT:
HA:
Quizzes-HA:
Periodical Examination: one
Project/Lab: one (under the guidance of the teachers the students will take a volume of poetry,
fiction or drama and write a term paper to show their understanding of it in a given context; sociological, psychological, historical, autobiographical etc.
Final Exam:
                              TOTAL NUMBER OF PERIODS: 45

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Students will be able to understand the relevance of literature in human life and appreciate
its aspects in developing finer sensibilities.

READINGS

1. An Introduction to the Study of English Literature, W.H. Hudson, Atlantic, 2007.
2. An Introduction to Literary Studies, Mario Klarer, Routledge, 2013.
3. The Experience of Poetry, Graham Mode, Open college of Arts with Open Unv Press,1991.
4. The Elements of Fiction: A Survey, Ulf Wolf (ed), Wolfstuff, 2114.
5. The Elements of Drama, J.L.Styan, Literary Licensing, 2011.

REFERENCES:

1. To be decided by the teacher and student, on the basis of individual student so as to enable him or her to write the term paper**.**

Course Code	FILM APPRECIATION	L	T	P	C
MC4303		3	0	0	0

COURSE OBJECTIVES:

1. In this course on film appreciation, the students will be introduced broadly to the development of film as an art and entertainment form. It will also discuss the language of cinema as it evolved over a century. The students will be taught as to how to read a film and appreciate the various nuances of a film as a text. The students will be guided to study film joyfully

UNIT I THEME - A: THE COMPONENT OF FILMS   **9**

A-1: The material and equipment A-2: The story, screenplay and script A-3: The actors, crew members, and the director A-4: The process of film making… structure of a film

UNIT II THEME - B: EVOLUTION OF FILM LANGUAGE   **9**

B-1: Film language, form, movement etc. B-2: Early cinema… silent film (Particularly French) B-3: The emergence of feature films: Birth of a Nation B-4: Talkies

UNIT III THEME - C: FILM THEORIES AND CRITICISM/APPRECIATION   **9**

C-1: Realist theory; Auteurists C-2: Psychoanalytic, Ideological, Feminists C-3: How to read films? C-4: Film Criticism / Appreciation

UNIT IV THEME – D: DEVELOPMENT OF FILMS   **9**

D-1: Representative Soviet films, D-2: Representative Japanese films, D-3: Representative Italian films, D-4: Representative Hollywood film and the studio system

UNIT V THEME - E: INDIAN FILMS   **9**

E-1: The early era, E-2: The important films made by the directors, E-3: The regional films, E-4: The documentaries in India

TOTAL NUMBER OF PERIODS: 45

READINGS

1. A Reader containing important articles on films will be prepared and given to the students. The students must read them and present in the class and have discussion on these.

Course Code	DISASTER RISK REDUCTION AND	L	T	P	C
MC4304	MANAGEMENT	3	0	0	0

COURSE OBJECTIVES:

1. To impart knowledge on concepts related to disaster, disaster risk reduction, disaster management
2. To acquaint with the skills for planning and organizing disaster response.

UNIT I HAZRADS, VULNERABILITY AND DISASTER RISKS   **9**

Definition: Disaster, Hazard, Vulnerability, Resilience, Risks – Types of Disasters: Natural, Human induced, Climate change induced –Earthquake, Landslide, Flood, Drought, Fire etc – Technological disasters- Structural collapse, Industrial accidents, oil spills -Causes, Impacts including social, Economic, political, environmental, health, psychosocial, etc.- Disaster vulnerability profile of India and Tamil Nadu - Global trends in disasters: urban disasters, pandemics, Complex emergencies, - Inter relations between Disasters and Sustainable development Goals

UNIT II DISASTER RISK REDUCTION (DRR)   **9**

Sendai Framework for Disaster Risk Reduction, Disaster cycle - Phases, Culture of safety, prevention, mitigation and preparedness community Based DRR, Structural- non-structural measures, Roles and responsibilities of- community, Panchayati Raj Institutions / Urban Local Bodies (PRIs/ULBs), States, Centre, and other stakeholders- Early Warning System – Advisories from Appropriate Agencies. - Relevance of indigenous Knowledge, appropriate technology and Local resources.

UNIT III DISASTER MANAGEMENT   **9**

Components of Disaster Management – Preparedness of rescue and relief, mitigation, rehabilitation and reconstruction- Disaster Risk Management and post disaster management – Compensation and Insurance- Disaster Management Act (2005) and Policy - Other related policies, plans, programmers and legislation - Institutional Processes and Framework at State and Central Level- (NDMA –SDMA-DDMA-NRDF- Civic Volunteers)

UNIT IV TOOLS AND TECHNOLOGY FOR DISASTER MANAGEMENT   **9**

Early warning systems -Components of Disaster Relief: Water, Food, Sanitation, Shelter, Health, Waste Management, Institutional arrangements (Mitigation, Response and Preparedness, – Role of GIS and Information Technology Components in Preparedness, Risk Assessment, Response and Recovery Phases of Disaster – Disaster Damage Assessment. - Elements of Climate Resilient Development –Standard operation Procedure for disaster response – Financial planning for disaster Management

UNIT V DISASTER MANAGEMENT: CASE STUDIES   **9**

Discussion on selected case studies to analyse the potential impacts and actions in the contest of disasters-Landslide Hazard Zonation: Earthquake Vulnerability Assessment of Buildings and Infrastructure: Case Studies, Drought Assessment: Case Studies, Coastal Flooding: Storm Surge Assessment, Floods: Fluvial and Pluvial Flooding: Case Studies; Forest Fire: Case Studies, Man Made disasters: Case Studies, Space Based Inputs for Disaster Mitigation and Management and field works related to disaster management.- Field work-Mock drill –

COURSE OUTCOME:

On completion of this course, the students will be able to:

CO1: To impart knowledge on the concepts of Disaster, Vulnerability and Disaster Risk reduction (DRR)

CO2: To enhance understanding on Hazards, Vulnerability and Disaster Risk Assessment prevention and risk reduction

CO3: To develop disaster response skills by adopting relevant tools and technology

CO4: Enhance awareness of institutional processes for Disaster response in the country and

CO5: Develop rudimentary ability to respond to their surroundings with potential Disaster response in areas where they live, with due sensitivity

TEXT BOOKS

1. Taimpo (2016), Disaster Management and Preparedness, CRC Publications
2. Singh R (2017), Disaster Management Guidelines for earthquakes, Landslides, Avalanches and tsunami, Horizon Press Publications
3. Singhal J.P. "Disaster Management", Laxmi Publications, 2010. ISBN-10: 9380386427 ISBN-13: 978-9380386423
4. Tushar Bhattacharya, "Disaster Science and Management", McGraw Hill India
5. Education Pvt. Ltd., 2012. ISBN-10: 1259007367, ISBN-13: 978-1259007361]

REFERENCE BOOKS

1. Govt. of India: Disaster Management Act, Government of India, New Delhi, 2005.
2. Government of India, National Disaster Management Policy, 2009.
3. Shaw R (2016), Community based Disaster risk reduction, Oxford University Press

Course Code	ENVIRONMENTAL SCIENCES AND	L	T	P	C
MC4406	SUSTAINABILITY	3	0	0	0

COURSE OBJECTIVES:

1. To introduce the basic concepts of environment, ecosystems and biodiversity and emphasize on the biodiversity of India and its conservation.

2. To impart knowledge on the causes, effects and control or prevention measures of environmental pollution and natural disasters.

3. To facilitate the understanding of global and Indian scenario of renewable and non-renewable resources, causes of their degradation and measures to preserve them.

4. To familiarize the concept of sustainable development goals and appreciate the interdependence of economic and social aspects of sustainability, recognize and analyse climate changes, concept of carbon credit and the challenges of environmental management.

5. To inculcate and embrace sustainability practices and develop a broader understanding on green materials, energy cycles and analyze the role of sustainable urbanization.

UNIT I ENVIRONMENT AND BIODIVERSITY   6

Definition, scope and importance of environment – need for public awareness. Eco-system and Energy flow– ecological succession. Types of biodiversity: genetic, species and ecosystem diversity– values of biodiversity, India as a mega-diversity nation – hot-spots of biodiversity – threats to biodiversity: habitat loss, poaching of wildlife, man-wildlife conflicts – endangered and endemic species of India – conservation of biodiversity: In-situ and ex-situ.

UNIT II ENVIRONMENTAL   POLLUTION   6

Causes, Effects and Preventive measures of Water, Soil, Air and Noise Pollutions. Solid, Hazardous and E-Waste management. Case studies on Occupational Health and Safety Management system (OHASMS). Environmental protection, Environmental protection acts .

UNIT III RENEWABLE SOURCES OF ENERGY   6

Energy management and energy conservation, New Energy Sources: Need of new energy sources. Different types of new energy sources. Applications of–Hydrogen energy resources, Tidal energy conversion. Concept, origin and power plants of geothermal energy.

UNIT IV SUSTAINABILITY AND MANAGEMENT   6

Development , GDP ,Sustainability- concept, needs and challenges-economic, social and aspects of sustainability-from unsustainability to sustainability-millennium development goals, and protocols-Sustainable Development Goals-targets, indicators and intervention areas Climate change- Global, Regional and local environmental issues and possible solutions-case studies. Concept of Carbon Credit, Carbon Footprint. Environmental management in industry-A case study.

UNIT V SUSTAINABILITY PRACTICES 6

Zero waste and R concept, Circular economy, ISO 14000 Series, Material Life cycle assessment, Environmental Impact Assessment. Sustainable habitat: Green buildings, Green materials, Energy efficiency, Sustainable transports. Sustainable energy: Non-conventional Sources, Energy Cycles- carbon cycle, emission and sequestration, Green Engineering: Sustainable urbanization- Socio- economical and technological change.

TOTAL NUMBER OF PERIODS: 30

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   To recognize and understand the functions of environment, ecosystems and biodiversity and their conservation.

CO2   To identify the causes, effects of environmental pollution and natural disasters and contribute to the preventive measures in the society.

CO3   To identify and apply the understanding of renewable and non-renewable resources and contribute to the sustainable measures to preserve them for future generations.

CO4   To recognize the different goals of sustainable development and apply them for suitable technological advancement and societal development.

CO5   To demonstrate the knowledge of sustainability practices and identify green materials, energy cycles and the role of sustainable urbanization.

TEXTBOOKS:

1. Anubha Kaushik and C. P. Kaushik's "Perspectives in Environmental Studies", 6th Edition, New Age International Publishers ,2018.
2. Benny Joseph, 'Environmental Science and Engineering', Tata McGraw-Hill, New Delhi, 2016.
3. Gilbert M.Masters, 'Introduction to Environmental Engineering and Science', 2nd edition, Pearson Education, 2004.
4. Allen, D. T. and Shonnard, D. R., Sustainability Engineering: Concepts, Design and Case Studies, Prentice Hall.
5. Bradley. A.S; Adebayo, A.O., Maria, P. Engineering applications in sustainable design and development, Cengage learning.
6. Environment Impact Assessment Guidelines, Notification of Government of India, 2006.
7. Mackenthun, K.M., Basic Concepts in Environmental Management, Lewis Publication, London, 1998.

REFERENCES:

1. R.K. Trivedi, 'Handbook of Environmental Laws, Rules, Guidelines, Compliances and Standards', Vol. I and II, Enviro Media. 38 . edition 2010.
2. Cunningham, W.P. Cooper, T.H. Gorhani, 'Environmental Encyclopedia', Jaico Publ., House, Mumbai, 2001.
3. Dharmendra S. Sengar, 'Environmental law', Prentice hall of India PVT. LTD, New Delhi, 2007.
4. Rajagopalan, R, 'Environmental Studies-From Crisis to Cure', Oxford University Press, Third Edition, 2015.
    Erach Bharucha "Textbook of Environmental Studies for Undergraduate Courses" Orient

Blackswan Pvt. Ltd. 2013.

APPENDIX C: MANDATORY COURSES II

Course Code	**WELL-BEING WITH TRADITIONAL**	**L**	**T**	**P**	**C**
MC4401	**PRACTICES, AYURVEDA AND SIDDHA**	3	0	0	0

COURSE OBJECTIVES:

1. To enjoy life happily with fun-filled new style activities that help to maintain health also
2. To adapt a few lifestyle changes that will prevent many health disorders
3. To be cool and handbill every emotion very smoothly in every walk of life
4. To learn to eat cost-effective but healthy foods that are rich in essential nutrients
5. To develop immunity naturally that will improve resistance against many health disorders

UNIT I HEALTH AND ITS IMPORTANCE   **2+4**

Health: Definition - Importance of maintaining health - More importance on prevention than treatment Ten types of health one has to maintain - Physical health - Mental health - Social health - Financial health - Emotional health - Spiritual health - Intellectual health - Relationship health - Environmental health - Occupational/Professional health.

Present health status - The life expectancy-present status - mortality rate - dreadful diseases - Non-communicable diseases (NCDs) the leading cause of death - 60% - heart disease – cancer – diabetes - chronic pulmonary diseases - risk factors – tobacco – alcohol - unhealthy diet - lack of physical activities.

Types of diseases and disorders - Lifestyle disorders – Obesity – Diabetes - cardiovascular diseases – Cancer – Strokes – COPD - Arthritis - Mental health issues.

Causes of the above diseases/disorders - Importance of prevention of illness - Takes care of health - Improves quality of life - Reduces absenteeism - Increases satisfaction - Saves time.

Simple lifestyle modifications to maintain health - Healthy Eating habits (Balanced diet according to age) Physical Activities (Stretching exercise, aerobics, resisting exercise) - Maintaining importance and actions to be taken

UNIT II DIET   **9**
Role of diet in maintaining health - energy one needs to keep active throughout the day - nutrients
one needs for growth and repair - helps one to stay strong and healthy - helps to prevent diet-related illness, such as some cancers - keeps active and - helps one to maintain a healthy

weight - helps to reduce risk of developing lifestyle disorders like diabetes – arthritis – hypertension – PCOD – infertility – ADHD – sleeplessness -helps to reduce the risk of heart diseases - keeps the teeth and
bones strong.

Balanced Diet and its 7 Components - Carbohydrates – Proteins – Fats – Vitamins – Minerals - Fibre and Water.

Food additives and their merits & demerits - Effects of food additives - Types of food additives - Food additives and processed foods - Food additives and their reactions

Definition of BMI and maintaining it with diet
Importance - Consequences of not maintaining BMI - different steps to maintain optimal BM

Common cooking mistakes
Different cooking methods, merits and demerits of each method.

UNIT III ROLE OF AYURVEDA & SIDDHA SYSTEMS IN MAINTAINING HEALTH   4+4

AYUSH systems and their role in maintaining health - preventive aspect of AYUSH - AYUSH as a soft therapy.

Secrets of traditional healthy living - Traditional Diet and Nutrition - Regimen of Personal and Social Hygiene - Daily routine (Dinacharya) - Seasonal regimens (Ritucharya) - basic sanitation and healthy living environment - Sadvritta (good conduct) - for conducive social life.

Principles of Siddha & Ayurveda systems - Macrocosm and Microcosm theory - Pancheekarana Theory / (Five Element Theory) 96 fundamental Principles - Uyir Thathukkal (Tri-Dosha Theory) - Udal Thathukkal

Prevention of illness with our traditional system of medicine Primary Prevention - To decrease the number of new cases of a disorder or illness - Health promotion/education, and - Specific protective measures - Secondary Prevention - To lower the rate of established cases of a disorder or illness in the population (prevalence) - Tertiary Prevention - To decrease the amount of disability associated with an existing disorder.

UNIT IV MENTAL WELLNESS   3+4
Emotional health - Definition and types - Three key elements: the subjective experience - the physiological response - the behavioural response - Importance of maintaining emotional health - Role of emotions in daily life -Short term and long-term effects of emotional disturbances - Leading a healthy life with emotions - Practices for emotional health - Recognize how thoughts influence emotions - Cultivate positive thoughts - Practice self-compassion - Expressing a full range of emotions.

Stress management - Stress definition - Stress in daily life - How stress affects one's life - Identifying the cause of stress - Symptoms of stress - Managing stress (habits, tools, training, professional help) - Complications of stress mismanagement.

Sleep - Sleep and its importance for mental wellness - Sleep and digestion.

Immunity - Types and importance - Ways to develop immunity

UNIT V YOGA   9

Definition and importance of yoga - Types of yoga - How to Choose the Right Kind for Individuals according to their age - The Eight Limbs of Yoga - Simple yogasan as for cure and prevention of health disorders - What yoga can bring to our life.

TOTAL NUMBER OF PERIODS: 45

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Learn the importance of different components of health
CO2   Gain confidence to lead a healthy life
CO3   Learn new techniques to prevent lifestyle health disorders
CO4   Understand the importance of diet and workouts in maintaining health

TEXTBOOKS:

1. Nutrition and Dietetics - Ashley Martin, Published by White Word Publications,
    New York, NY 10001, USA
2. Yoga for Beginners_ 35 Simple Yoga Poses to Calm Your Mind and Strengthen Your Body,
    by Cory Martin, Copyright © 2015 by Althea Press, Berkeley, California

REFERENCES:

1. WHAT WE KNOW ABOUT EMOTIONAL INTELLIGENCE How It Affects Learning, Work,
    Relationships, and Our Mental Health, by Moshe Zeidner, Gerald Matthews, and Richard D.
    Roberts
2. A Bradford Book, The MIT Press, Cambridge, Massachusetts, London, England The
    Mindful Self-Compassion Workbook, Kristin Neff, Ph.D Christopher Germer, Ph.D, Published
    by The Guilford Press A Division of Guilford Publications, Inc.370 Seventh Avenue, Suite
    1200, New York, NY 10001.

WEB REFERENCE:

1. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4799645/
2. Simple lifestyle modifications to maintain health
3. https://www.niddk.nih.gov/health-information/diet-nutrition/changing-habits-better-
4. health#:~:text=Make%20your%20new%20healthy%20habit,t%20have%20time%20to%2
    0cook.
5. Read more: https://www.legit.ng/1163909-classes-food-examples-functions.html
6. https://www.yaclass.in/p/science-state-board/class-9/nutrition-and-health-5926
7. Benefits of healthy eating https://www.cdc.gov/nutrition/resources-publications/benefits-
    ofhealthy-eating.html
8. Food   additives
    https://www.betterhealth.vic.gov.au/health/conditionsandtreatments/foodadditives
9. BMI https://www.hsph.harvard.edu/nutritionsource/healthy-weight/
10. https://www.who.int/europe/news-room/fact-sheets/item/a-healthy-lifestyle---
    whorecommendations
11. Yoga https://www.healthifyme.com/blog/types-of-yoga/
12. https://yogamedicine.com/guide-types-yoga-styles/
13. Ayurveda   :   https://vikaspedia.in/health/ayush/ayurveda-1/concept-of-healthy-living-in-
    ayurveda
14. Siddha : http://www.tkdl.res.in/tkdl/langdefault/Siddha/Sid_Siddha_Concepts.asp
15. CAM : https://www.hindawi.com/journals/ecam/2013/376327/
16. Preventive herbs : https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3847409

Course Code	**HISTORY OF SCIENCE AND TECHNOLOGY IN INDIA**	L	T	P	C
MC4402		3	0	0	0

UNIT I CONCEPTS AND PERSPECTIVES   9

Meaning of History Objectivity, Determinism, Relativism, Causation, Generalization in History; Moral judgment in history

Extent of subjectivity, contrast with physical sciences, interpretation and speculation, causation

verses evidence, the concept of historical inevitability, and Historical Positivism.

Science and Technology-Meaning, Scope and Importance, Interaction of science, technology & society, Sources of history on science and technology in India

UNIT II HISTORIOGRAPHY OF SCIENCE AND TECHNOLOGY IN INDIA   9

Introduction to the works of D.D. Kosambi, Dharmpal, Debiprasad Chattopadhyay, Rehman, S. Irfan Habib, Deepak Kumar, Dhruv Raina, and others.

UNIT III SCIENCE AND TECHNOLOGY IN ANCIENT INDIA   9

Technology in pre-historic period Beginning of agriculture and its impact on technology Science and Technology during Vedic and Later Vedic times Science and technology from 1st century AD to C-1200

UNIT IV SCIENCE AND TECHNOLOGY IN MEDIEVAL INDIA   9

Science and the Empire
Indian response to Western Science
Growth of techno-scientific institutions

UNIT V SCIENCE AND TECHNOLOGY IN A POST-INDEPENDENT INDIA   9

Science, Technology and Development discourse
Shaping of the Science and Technology Policy
Developments in the field of Science and Technology
Science and technology in globalizing India
Social implications of new technologies like the Information Technology and Biotechnology
.

TOTAL NUMBER OF PERIODS: 45

REFERENCES:

1. Acharya , P. K ., Dictionary o f Hindu Architecture. London, 1927.
2. Bose , D. M ., Sen , S. N., and Subba rayappa , B. V. (Eds.), A Concise History o f Science in India. Indian National Science Academy, New Delhi, 1971.
3. Chatterji, Sunm Kumar (E d.), The Cultural Heritage o f India. Vol. V. The Ramakrishna Mission Institute of Culture, Calcutta, 1978.
4. Chattopadhyaya , Debiprasad (Ed.), Studies in the History o f Science in India (2 Vols.). Editorial Enterprises, New Delhi, 1982.
5. Dampier, W. C., History of Science and its Relations with Philosophy and Religion. 4th Edn. Cam bridge, 1961.
6. Forbes , George , History o f Astronomy. Watts & Co., London, 1909.
7. Forbes , R . J ., Metallurgy in Antiquity. Leyden, 1950.
8. Haldane , J . B. S., Science and Indian Culture. New Age Publishers Pvt. L td., Calcutta, 1965.
9. Frawley, D., Planets in the Vedic literature, Indian Journal of History of Science. 29.495-506, 1994. 10 B Datta and A N Singh, History of Hindu Mathematics: A source book, Parts 1 and 2 (single volume), Asia Publishing House, Bombay, 1962.
10. George G Joseph, Crest of the Peacock, Non-European roots of mathematics, Third edition,

Course Code	POLITICAL AND ECONOMIC THOUGHT FOR A HUMANE	L	T	P	C
MC4403	SOCIETY	3	0	0	0

COURSE OBJECTIVES:

1. This course will begin with a short overview of human needs and desires and how different political-economic systems try to fulfil them. In the process, we will end with a critique of different systems and their implementations in the past, with possible future directions.

UNIT I INTRODUCTION   **9**

Considerations for humane society, holistic thought, human being's desires, harmony in self, harmony in relationships, society, and nature, societal systems.

(Refs: A Nagaraj, M K Gandhi, JC Kumarappa)

UNIT II CAPATALISM AND MONOPOLIES   **7**

Capitalism – Free markets, demand-supply, perfect competition, laissez-faire, monopolies, imperialism. Liberal democracy. (Refs: Adam smith, J S Mill)

Fascism and totalitarianism. World War I and II. Cold war.

UNIT III COMMUNISM   **5**

Communism – Mode of production, theory of labour, surplus value, class struggle, dialectical materialism, historical materialism, Russian and Chinese models. (Refs: Marx, Lenin, Mao, M N Roy)

UNIT IV HUMAN EMPOWERMENT   **9**

Welfare state. Relation with human desires. Empowered human beings, satisfaction.
Gandhian thought. Swaraj, Decentralized economy & polity, Community. Control over one's lives.
Relationship with nature.   (Refs: M K Gandhi, Schumacher, Kumarappa)

UNIT V EDUCATION AND TECHNOLOGY   **6**

Essential elements of Indian civilization. (Refs: Pt Sundarlal, R C Mazumdar, Dharampal)

Technology as a driver of society, Role of education in shaping of society. Future directions. (4 lectures) (Refs: Nandakishor Acharya, David Dixon, Levis Mumford)

        TOTAL NUMBER OF PERIODS: **39**

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   The students will get an understanding of how societies are shaped by philosophy, political and economic system, how they relate to fulfilling human goals & desires with some case studies of how different attempts have been made in the past and how they have fared.

REFERENCES:

Authors mentioned along with topics above. Detailed reading list will be provided

Course Code	STATE, NATION BUILDING AND POLITICS IN INDIA	L	T	P	C
MC4404		3	0	0	0

COURSE OBJECTIVES:

1. The objective of the course is to provide an understanding of the state, how it works through its main organs, primacy of politics and political process, the concept of sovereignty and its changing contours in a globalized world. In the light of this, an attempt will be made to acquaint the students with the main development and legacies of national movement and constitutional development in India, reasons for adopting a Parliamentary-federal system, the broad philosophy of the Constitution of India and the changing nature of Indian Political System. Challenges/ problems and issues concerning national integration and nation-building will also be discussed in the contemporary context with the aim of developing a future vision for a better India.

Understanding the need and role of State and politics. Development of Nation-State, sovereignty, sovereignty in a globalized world. Organs of State – Executive, Legislature, Judiciary. Separation of powers, forms of government unitary-federal, Presidential-Parliamentary, The idea of India.

1857 and the national awakening.

1885 Indian National Congress and development of national movement – its legacies. Constitution making and the Constitution of India. Goals, objective and philosophy. Why a federal system? National integration and nation-building.

Challenges of nation-building – State against democracy (Kothari) New social movements. The changing nature of Indian Political System, the future scenario. What can we do?

TOTAL NUMBER OF PERIODS: 45

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   It is expected that this course will make students aware of the theoretical aspect of the state, its organs, its operationalization aspect, the background and philosophy behind the founding of the present political system, broad streams and challenges of national integration and nation-building in India. It will equip the students with the real understanding of our political system/ process in correct perspective and make them sit up and think for devising ways for better participation in the system with a view to making the governance and delivery system better for the common man who is often left unheard and unattended in our democratic setup besides generating a lot of dissatisfaction and difficulties for the system.

REFERENCES:

1. Sunil Khilnani, The Idea of India. Penguin India Ltd., New Delhi.

2. Madhav Khosla, The Indian Constitution, Oxford University Press. New Delhi, 2012.

3. Brij Kishore Sharma, Introduction to the Indian Constitution, PHI, New Delhi, latest

edition.

4. Sumantra Bose, Transforming India: Challenges to the World's Largest Democracy,

5. Picador India, 2013.

6. Atul Kohli, Democracy and Discontent: India's Growing Crisis of Governability, Cambridge University Press, Cambridge, U. K., 1991.

7. M. P. Singh and Rekha Saxena, Indian Politics: Contemporary Issues and Concerns, PHI, New Delhi, 2008, latest edition.

8. Rajni Kothari, Rethinking Democracy, Orient Longman, New Delhi, 2005

Course Code	INDUSTRIAL SAFETY	L	T	P	C
MC4405		3	0	0	0

COURSE OBJECTIVES:

2. To Understand the Introduction and basic Terminologies safety.

3. To enable the students to learn about the Important Statutory Regulations and standards.

4. To enable students to Conduct and participate the various Safety activities in the industry.

5. To have knowledge about Workplace Exposures and Hazards.

6. To assess the various Hazards and consequences through various Risk Assessment Techniques.

UNIT I SAFETY TERMINOLOGIES   **9**

Hazard-Types of Hazard- Risk-Hierarchy of Hazards Control Measures-Lead indicators- lag Indicators-Flammability- Toxicity Time-weighted Average (TWA) - Threshold LimitValue (TLV) - Short Term Exposure Limit (STEL)- Immediately dangerous to life or health (IDLH)- acute and chronic Effects- Routes of Chemical Entry-Personnel Protective Equipment- Health and Safety Policy-Material Safety Data Sheet MSDS

UNIT II STANDARDS AND REGULATIONS   **9**

Indian Factories Act-1948- Health- Safety- Hazardous materials and Welfare- ISO 45001:2018 occupational health and safety (OH&S) - Occupational Safety and Health Audit IS14489:1998- Hazard Identification and Risk Analysis- code of practice IS 15656:2006

UNIT III SAFETY ACTIVITIES   **9**

Toolbox Talk- Role of safety Committee- Responsibilities of Safety Officers and Safety Representatives- Safety Training and Safety Incentives- Mock Drills- On-site Emergency Action Plan- Off-site Emergency Action Plan- Safety poster and Display- Human Error Assessment

UNIT IV WORKPLACE HEALTH AND SAFETY   **9**

Noise hazard- Particulate matter- musculoskeletal disorder improper sitting poster and lifting Ergonomics RULE & REBA- Unsafe act & Unsafe Condition- Electrical Hazards- Crane SafetyToxic gas Release.

UNIT V HAZARD IDENTIFICATION TECHNIQUES   **9**

Job Safety Analysis-Preliminary Hazard Analysis-Failure mode and Effects Analysis- Hazard and
Operability- Fault Tree Analysis- Event Tree Analysis Qualitative and Quantitative Risk Assessment Checklist Analysis- Root cause analysis- What-If Analysis- and Hazard Identification and Risk Assessment

                 TOTAL NUMBER OF PERIODS: **45**

COURSE OUTCOMES:

On completion of this course, the students will be able to:

CO1   Understand the basic concept of safety.
CO2   Obtain knowledge of Statutory Regulations and standards.
CO3   Know about the safety Activities of the Working Place.
CO4   Analyse on the impact of Occupational Exposures and their Remedies
CO5   Obtain knowledge of Risk Assessment Techniques.

TEXTBOOKS:

1. R.K. Jain and Prof. Sunil S. Rao Industrial Safety, Health and Environment Management Systems KHANNA PUBLISHER
2. L. M. Deshmukh Industrial Safety Management: Hazard Identification and Risk Control McGraw-Hill Education

REFERENCES:

1. Frank Lees (2012) 'Lees' Loss Prevention in Process Industries.Butterworth-Heinemann publications, UK, 4th Edition.
2. John Ridley & John Channing (2008)Safety at Work: Routledge, 7th Edition.
3. Dan Petersen (2003) Techniques of Safety Management: A System Approach.
4. Alan Waring.(1996).Safety management system: Chapman &Hall,England Society of Safety Engineers, USA

WEB REFERENCE:

1. ISO 45001:2018 occupational health and safety (OH&S) International Organization for Standardization https://www.iso.org/standard/63787.html
2. Indian Standard code of practice on occupational safety and health audit https://law.resource.org/pub/in/bis/S02/is.14489.1998.pdf
3. Indian Standard code of practice on Hazard Identification and Risk Analysis IS 15656:2006 https://law.resource.org/pub/in/bis/S02/is.15656.2006.pdf

*****