Curriculum for UG Degree Course in

Artificial Intelligence and Data Science

Regulation 2024

CHENNAI INSTITUTE OF TECHNOLOGY

(Autonomous)

Document Version

Version Number	Date	Author	Major Updates	Approved by
1	30.03.2024	Dr.S. Veeramalai	Font size, Font Style and Structure	
2	24.06.2024	Dr.S. Veeramalai	Discrete Mathematics, Open Electives and Mandatory Course	
3	07.08.2024	Dr.S. Veeramalai	Course code and Course Credit has been checked and finalized.	

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)	13
2	Basic Science Courses (BSC)	16
3	Engineering Science Courses (ESC)	12
4	Program Core Courses (PCC)	83
5	Professional Elective Courses (PEC)	18
6	Open Elective Courses (OEC)	6
7	Employability Enhancement Skills (EES)	21
8	Mandatory Course (MC)	0
	TOTAL	**169**

C. Course code and definition

Code	Definition
L	Lecture
T	Tutorial
P	Practical
C	Credits
<AD>	Professional core courses
<AD> PE	Professional Elective courses
<AD> OE	Open Elective Courses
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 "4" for 2024 Regulation) Digit at hundred's place signifies the semester in which course is offered. The last two digits represent the serial order of course within the semester. For example, 4101, 4102, … are courses offered during the first semester.

D. Category-wise Courses

Humanities & Social Science Courses (HSMC)

S. No	Course Tile	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	Professional Ethics & Human Values	VII	3-0-0-3
	Total Credits		**13**

Basic Science Courses (BSC)

S. No	Course Tile	Semester	L-T-P-C
1	Mathematics – I (Calculus and Linear Algebra)	I	3-1-0-4
2	Engineering Physics	I	3-1-2-4
3	Mathematics -II (Probability and Statistics)	II	3-1-0-4
4	Discrete Mathematics	III	3-1-0-4
		Total Credits	**16**

Engineering Science Courses (ESC)

S. No	Course Tile	Semester	L-T-P-C
1	C++ Programming	I	3-0-2-4
2	Web Development Essentials	I	3-0-2-4
3	Digital Design and Computer Organization	II	3-0-2-4
		Total Credits	**12**

Program Core Courses (PCC)

S. No	Course Title	SEMESTER	L-T-P-C
1	Artificial Intelligence	I	3-0-2-4
2	Data Structures Using C++	II	3-0-2-4
3	Database Management Systems	II	3-0-2-4
4	Introduction to Java Programming	II	3-0-2-4
5	Operating System	III	3-0-2-4
6	Design and Analysis of Algorithms	III	3-0-2-4
7	Web Development Frameworks and Practices	III	3-0-2-4
8	Data Science and Visualization	III	3-0-2-4
9	Python for Parallel Processing	III	3-0-2-4
10	Core Course Project -I	III	0-0-2-1
11	Computing Theory and Compiler Design	IV	3-0-0-3
12	Big Data Analytics	IV	3-0-2-4
13	Machine Learning	IV	3-0-2-4
14	Data Warehousing and Data Mining	IV	3-0-2-4
15	Artificial Intelligence of Things (AIoT)	IV	3-0-0-3
16	Computer Networks	IV	3-0-2-4
17	Core Course Project -II	IV	0-0-2-1

18	Natural Language Processing	V	3-0-2-4
19	Deep Learning	V	3-0-2-4
20	Reinforcement Learning	V	2-0-2-3
21	Core Course Project - III	V	0-0-2-1
22	Cloud and Distributed Computing	VI	3-0-0-3
23	Generative AI	VI	3-0-2-4
24	Computer Vision	VI	2-0-2-3
25	Core Course Project - IV	VI	0-0-2-1
		Total Credits	**83**

Professional Elective courses (PE)

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

Employability Enhancement Skills (EES)

S. No	Course Title	Semester	L-T-P-C
1	Employability Enhancement Skills-I	I	0-0-2-1
2	Employability Enhancement Skills-II	II	0-0-2-1
3	Employability Enhancement Skills-III	III	0-0-2-1
4	Employability Enhancement Skills-IV	IV	0-0-2-1
5	Employability Enhancement Skills-V	V	0-0-2-1
6	Internship	VII	0-0-8-4
7	Project work Phase - I	VII	0-0-12-6
	Project work Phase - II	VIII	0-0-12-6
		Total Credits	**21**

Mandatory Course (MC)

S. No	Course Title	Semester	L-T-P-C
1	Introduction to Women and Gender Studies	III	2-0-0-0
2	Elements for Literature	III	2-0-0-0

3	Film Appreciation	III	2-0-0-0
4	Disaster Management	III	2-0-0-0
5	Design Thinking	III	2-0-0-0
6	Environmental Science and Sustainability	IV	2-0-0-0
7	Well Being with Traditional practices (Yogam Ayurveda and Siddha)	IV	2-0-0-0
8	History of Science and Technology in India	IV	2-0-0-0
9	Political and Economic Thought for a Humane Society	IV	2-0-0-0
10	State, Nation building and Politics in India	IV	2-0-0-0
11	Industrial Safety	IV	2-0-0-0
		Total Credits	**0**

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 to familiarizing 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 the 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 the 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 the 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) in the campus and learn additional technical skills
b. Students should be encouraged to participate in internal / external competitions, hackathons, etc. on a regular basis

Section 2: Semester-wise Structure and Curriculum for UG Course in B.Tech. Artificial Intelligence and Data Science

Semester I							
S. No	**Theory/ Practical / T&P**	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	Theory	IP4100	Induction Programme	-	-	-	-
2	Theory	MA4101	Mathematics – I Calculus and Linear Algebra	3	1	0	4
3	Theory	HS4101	தமிழர் மரபு / Heritage of Tamils	1	0	0	1
4	T&P	HS4102	Communicative English - I	3	0	2	4
5	T&P	PH4101	Engineering Physics	3	0	2	4
6	T&P	AD4101	Artificial Intelligence	3	0	2	4
7	T&P	CS4101	C++ Programming	3	0	2	4
8	T&P	EC4112	Digital Design and Computer Organization	3	0	2	4
9	Practical	ES4101	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	MA4201	Mathematics -II Probability and Statistics	3	1	0	4
2	Theory	HS4201	தமிழரும் தொழில் நுட்பமும் / Tamils and Technology	1	0	0	1
3	T&P	HS4202	Communicative English – II	3	0	2	4
4	T&P	CS4201	Web Development Essentials	3	0	2	4
5	T&P	CS4202	Data Structures Using C++	3	0	2	4
6	T&P	CS4203	Database Management Systems	3	0	2	4
7	T&P	CS4204	Introduction to Java Programming	3	0	2	4
8	Practical	ES 4201	Employability Enhancement Skills-II	0	0	2	1
9			NCC Credit Course Level 1 ARMY WING*	2	0	0	2
						Total	**26**

* NCC Credit Course is offered for NCC students only. The grades earned by the students will be recorded in the Mark Sheet, however the same shall not be considered for the computation of CGPA

Semester III							
S. No	**Theory/ Practical / T&P**	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	Theory	MA4302	Discrete Mathematics	3	1	0	4
2	T&P	AD4301	Operating System	3	0	2	4
3	T&P	AD4302	Design and Analysis of Algorithms	3	0	2	4
4	T&P	AD4303	Web Development Frameworks and Practices	3	0	2	4
5	T&P	AD4304	Data Science and Visualization	3	0	2	4
6	T&P	AD4305	Parallel Processing through Python	3	0	2	4
7	Practical	ES4301	Employability Enhancement Skills-III	0	0	2	1
8	Practical	AD4306	Core Course Project -I	0	0	2	1
9	Theory	MC43XX	Mandatory Course – I	2	0	0	0
						Total	**26**

Semester IV							
S. No	**Theory/ Practical / T&P**	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	Theory	AD4401	Computing Theory and Compiler Design	3	0	0	3
2	T&P	AD4402	Big Data Analytics	3	0	2	4
3	T&P	AD4403	Machine Learning	3	0	2	4
4	T&P	AD4404	Data Warehousing and Data Mining	3	0	2	4
5	Theory	AD4405	Artificial Intelligence of Things (AIoT)	3	0	0	3
6	T&P	AD4406	Computer Networks	3	0	2	4
7	Practical	ES4401	Employability Enhancement Skills-IV	0	0	2	1
8	Practical	AD4407	Core Course Project - II	0	0	2	1
9	Theory	MC44XX	Mandatory Course – II	2	0	0	0
			NCC Credit Course Level 2 NAVAL WING*	2	0	0	2
						Total	**24**

* NCC Credit Course is offered for NCC students only. The grades earned by the students will be recorded in the Mark Sheet, however the same shall not be considered for the computation of CGPA

Semester V							
S. No	**Theory/ Practical / T&P**	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	T&P	AD4501	Natural Language Processing	3	0	2	4
2	T&P	AD4502	Deep Learning	3	0	2	4
3	T&P	AD4503	Reinforcement Learning	2	0	2	3
4	T&P		Professional Elective-I	2	0	2	3
5	T&P		Professional Elective-II	2	0	2	3
6	T&P		Professional Elective-III	2	0	2	3
7	Practical	ES4501	Employability Enhancement Skills-V	0	0	2	1
8	Practical	AD4504	Core Course Project - III	0	0	2	1
						Total	**22**

Semester VI							
S. No	**Theory/ Practical / T&P**	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	Theory	AD4601	Cloud and Distributed Computing	3	0	0	3
2	T&P	AD4602	Generative AI	3	0	2	4
3	T&P	AD4603	Computer Vision	2	0	2	3
4	T&P		Professional Elective - IV	2	0	2	3
5	T&P		Professional Elective - V	2	0	2	3
6	Theory		Open Elective I	3	0	0	3
7	Practical	AD4604	Core Course Project - IV	0	0	2	1
8			NCC Credit Course Level 3 AIR FORCE WING*	2	0	0	2
						Total	**20**

* NCC Credit Course is offered for NCC students only. The grades earned by the students will be recorded in the Mark Sheet, however the same shall not be considered for the computation of CGPA

Semester VII							
S. No	**Theory/ Practical / T&P**	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	Theory	HS4701	Professional Ethics and Universal Human Values	3	0	0	3
2	T&P		Professional Elective-VI	2	0	2	3
3	Theory		Open Elective-II	3	0	0	3
4	Practical	ES4701	Internship	0	0	8	4
5	Practical	ES4702	Project work Phase - I	0	0	12	6
						Total	**19**

Semester VIII							
S. No	**Theory/ Practical / T&P**	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	Practical	ES4801	Project work Phase - II	0	0	12	6
						Total	**6**

Semester I

Course Code	Mathematics – I	L	T	P	C
MA4101	Calculus and Linear Algebra	3	1	0	4

COURSE OBJECTIVES:

COURSE OBJECTIVES:

The main objectives of this course are to:

- To familiarize the students with differential calculus.
- To acquire the knowledge of evaluating integrals and their applications.
- To introduce the concept of ordinary differential equations in engineering problems.
- To develop the use of matrix algebra techniques for practical applications.
- To acquire the knowledge of vector spaces and linear transformation in all engineering disciplines.

UNIT I DIFFERENTIAL CALCULUS 12

Representation of a function - Limits - Continuity – Derivatives- Mean Value theorem - Rolle's theorem - Maclaurin series - Partial derivatives - Total derivatives - Taylors series - Maxima and minima.

UNIT II INTEGRAL CALCULUS 12

Multiple Integration: Double and Triple integrals - Change of order of integration in double integrals - Change of variables (Cartesian to polar) -Volume of solids - Gradient - Curl - Divergence - Theorems of Green in a plane - Gauss and Stokes theorems (Excluding Proof).

UNIT III DIFFERENTIAL EQUATIONS 12

First order linear and nonlinear differential equations - Higher order linear differential equations with constant coefficients - Method of Variation of Parameters - Cauchy's and Legendre's equations - Solution of partial differential equations.

UNIT IV LINEAR ALGEBRA 12

Matrices: Determinants - rank of a matrix - System of linear equations (Cramer's Rule, inverse of a matrix, Gauss elimination and Gauss-Jordan) - Eigen Values - Eigen Vectors - Reduce the quadratic form into Canonical form - LU decomposition - Singular Value Decomposition.

UNIT V VECTOR SPACES AND LINEAR TRANSFORMATIONS 12

Vector Space – Subspace - linear dependence and independence - bases and dimensions - Linear transformations - Null space - range - Dimension theorem - Matrix representation of a linear transformations.

TOTAL: 60 PERIODS

COURSE OUTCOMES:

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

CO1: Apply differential calculus tools in solving various applications in real situations.

CO2: Able to use the integral ideas in solving areas, volumes and other practical problems.

CO3: Apply various techniques in solving ordinary differential equations.

CO4: Recalling the matrix algebra methods for solving the practical problems.

CO5: Understand the concepts of vector spaces and applications of linear transformations.

TEXT BOOKS:

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.
3. Friedberg. A.H., Insel. A.J, and Spence. L ., "Linear Algebra", Prentice Hall of India, New Delhi, 4th Edition, 2004.

REFERENCES:

1. Weir, M.D and Joel Hass, "Thomas Calculus", Pearson India, 12thEdition, 2016.
2. Jain R.K. and Iyengar S.R.K., "Advanced Engineering Mathematics", Narosa Publications, New Delhi, 3rd Edition, 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, New Delhi, 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. Kumaresan. S., "Linear Algebra – A Geometric Approach", Prentice – Hall of India, New Delhi, Reprint, 2010.

TOTAL: 60 PERIODS

Course Code	HERITAGE OF TAMILS	L	T	P	C
HS4101		1	0	0	1

COURSE OBJECTIVES:

The main objectives of this course are to:

- Analyze Dravidian languages, particularly Tamil, and their literature's secular and distributive justice themes.
- Explore Tamil heritage from rock art to modern sculpture, emphasizing temple roles.
- Examine Tamil folk and martial arts' cultural significance.
- Investigate the Thinai concept and societal principles from ancient Tamil texts.
- Evaluate Tamil contributions to Indian culture, focusing on the freedom movement and cultural influence.

Course Description

Explore the Tamil language's evolution, classical literature's nuances, and the rich cultural heritage of Tamil Nadu. Delve into folk arts, martial traditions, and Tamil's contribution to Indian culture and freedom movements. Prerequisites include basic knowledge of Indian history and an interest in diverse cultural traditions.

Prerequisites

Basic understanding of Indian history and culture.
Familiarity with literary concepts and analysis.
Interest in exploring diverse cultural traditions.
Willingness to engage with both ancient texts and modern interpretations of Tamil culture and heritage.

UNIT I LANGUAGE AND LITERATURE **3**

Language Families in India – Dravidian Languages – Tamil as a Classical Language – Classical Literature in Tamil – Secular Nature of Sangam Literature – Distributive Justice in Sangam Literature – Management Principles in Thirukural – Tamil Epics and Impact of Buddhism & Jainism in Tamil Land – Bakthi Literature Azhwars and Nayanmars – Forms of minor Poetry – Development of Modern literature 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 Ports of Sangam Age – Export and Import during Sangam Age – Overseas Conquest of Cholas.

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

Contribution of Tamils to Indian Freedom Struggle – The Cultural Influence of Tamils over the other parts of India – Self–Respect Movement – Role of Siddha Medicine in Indigenous Systems of Medicine – Inscriptions & Manuscripts – Print History of Tamil Books.

15 PERIODS

Course Format

Lectures and discussions, Online resources and tutorials.
Assessments & Grading
Quizzes / Assignments, 3 Internal Assessments, Final Examination

COURSE OUTCOMES:

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

CO1: Understand Dravidian languages' significance, especially Tamil's classical literature.
CO2: Analyze Tamil art evolution from ancient to modern forms.
CO3: Explore Tamil folk arts, martial arts, and traditional sports.
CO4: Examine the Thinai concept encompassing flora, fauna, and moral values.
CO5: Evaluate Tamil contributions to Indian culture and freedom movement.

TEXT BOOKS:

1. Social Life of Tamils (Dr. K.K. Pillay) A joint publication of TNTB & ESC and RMRL.
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. Studies in the History of India with Special Reference to Tamil Nadu (Dr. K.K.Pillay) Published by The Author.

REFERENCES:

1. 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.
2. Porunai Civilization Jointly Published by Department of Archaeology & Tamil Nadu Text Book and Educational Services Corporation, Tamil Nadu.
3. Journey of Civilization Indus to Vaigai (R. Balakrishnan) Published by RMRL.

TOTAL: 15 PERIODS

Course Code	COMMUNICATIVE ENGLISH –I	L	T	P	C
HS4102		3	1	0	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 NewsSpeaking- Just A Minute, Introducing a friend

Reading- Reading Newspapers / Articles/ Magazines, Reading comprehension

Vocabulary – Synonyms& Antonyms, Acquaintance with Prefixes &suffixes from foreign languages in English to form derivatives, Word formation

Grammar – Parts of Speech, Mixed Tenses, Active &Passive VoiceWriting –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 / ConversationsSpeaking- 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:
Extempore (Oral)
Picture Description
Summarizing a TED talk.
Conversation on asking directions, about purchasing a product.
Role play.

TOTAL NUMBER OF PERIODS: 60

COURSE OUTCOMES:
Upon 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.

TEXT BOOKS:

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 b y 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
PH4101		3	0	2	4

COURSE OBJECTIVES:

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

UNIT I MECHANICS AND PROPERTIES OF MATTER **9**

Center of mass (CM) – motion of the CM – moment of inertia – theorems of M. I – moment of inertia of continuous bodies (Ring, Disc) – gyroscope.
Elasticity –Type of modulus: Young's Modulus, Bulk Modulus, Rigidity Modulus –Poisson ratio - 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 – Absorption Coefficient. -Doppler effect. Ultrasonic – Production of Ultrasonic 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 center – Einstein's coefficient – Principles of Laser – population inversion- Components Laser systems - Types of laser – Nd – YAG laser, CO2 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 – Fiber Optical Communication system (Block diagram) – Displacement sensors- Temperature/Pressure sensors –Optical fibers in computers - Medical Applications: Endoscope.

UNIT V QUANTUM MECHANICS **9**

Blackbody Radiation - 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: One Dimensional Box –Qubit- Quantum computing – the role of Quantum computing in advancing Artificial intelligence.

THEORY: 45 PERIODS

PRACTICAL EXERCISES (FIVE ONLY)

6. Ultrasonic interferometer – determination of the velocity of sound and compressibility of liquids.
7. Non-Uniform bending – Determination of Young's modulus.
8. Uniform bending – Determination of Young's modulus.
9. Laser – Determination of the wavelength of the laser using grating.

10. Optical fiber – Determination of Numerical Aperture and acceptance angle.
11. Acoustic grating – Determination of velocity of ultrasonic waves in liquids.
12. Michelson Interferometer.

PRACTICAL: 30 PERIODS

TOTAL: 75 PERIODS

COURSE OUTCOMES

After the completion of the course, 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. Acquire knowledge in laser and its applications
CO4. Demonstrate a strong foundational knowledge in fiber optics.
CO5. Comprehend and apply quantum mechanical principles.

TEXT BOOKS

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 (Indian Edition),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. Hugh D. Young, Roger A. Freedman, A. Lewis Ford, Searls and Zemansky University Physics, 2009.
5. David J. Griffiths, "Introduction to Quantum Mechanics", 2nd edition, Cambridge University Press, 2017.

WEB REFERENCES

13. Advanced Quantum Mechanics With Applications – https://nptel.ac.in/courses/115103104
14. Quantum Mechanics And Molecular Spectroscopy –

https://nptel.ac.in/courses/104101126

15. Fiber Optic Communication Technology – https://nptel.ac.in/courses/108106167
16. Introduction To Photonics – https://nptel.ac.in/courses/108106135
17. Introduction To Laser – https://nptel.ac.in/courses/115102124
18. Biomedical Ultrasound - https://nptel.ac.in/courses/121108458

Course Code	ARTIFICIAL INTELLIGENCE	L	T	P	C
AD4101		3	0	2	4

Course Objectives:

- To develop an understanding of the basic concepts of Artificial Intelligence.
- To analyze the nature of various advanced search strategies in AI.
- To impart knowledge representation, reasoning and planning mechanisms.
- To resolve Complex Decision-Making rules and their applications in AI.

UNIT: I Introduction to Artificial Intelligence **9**

Introduction: Foundations, Intelligent agents: Environments, structure of agents. Problem solving: searching for solutions, uninformed and informed search strategies, heuristic functions.

UNIT: II Advanced Search strategies **9**

Local Search Algorithms and Optimization Problems, Local search in Continuous Spaces, Nondeterministic Actions, unknown environments. Global optimization algorithms - Genetic Algorithms, Particle Swarm Optimization Algorithm, Ant Colony Optimization, Gravitational Search Algorithm. Constraint satisfaction problems (CSP): Backtracking and local search for CSPs. Adversarial Search: Games, Optimal decisions, alpha-beta pruning, imperfect real-time decisions, stochastic games, partially observables games.

UNIT: III Knowledge Representation **9**

Logical systems – Knowledge Based systems - Representing knowledge using Propositional Logic – Inference in Propositional Logic using Laws of Inference, Forward Chaining, Backward Chaining, Resolution. Representing knowledge using First Logic Order Logic Inference in First Order Logic using Unification, Forward Chaining, Backward Chaining, Ontologies and Classes.

UNIT: IV Planning and Quantifying Uncertainty **9**

Classical Planning: Definition, Algorithms for Planning as State-Space Search, Planning Graphs, Classical Planning Approaches. Planning and Acting: Time, Schedules, and Resources, Hierarchical Planning, Planning and Acting in Nondeterministic Domains, Multi-agent Planning.

Acting under Uncertainty, -Conditional Independence- Bayes Rule –Naïve Bayes Classifier - Bayesian Belief Network- Inference in Bayesian Belief Network- Making Decisions in Complex Environments- Markov Decision Processes.

UNIT - V Reinforced Learning and AI Applications **9**

Introduction, Bandit algorithms – UCB, PAC, Bandit algorithms –Median Elimination, Policy Gradient
Full RL & MDPs, Bellman Optimality, Dynamic Programming & TD Methods. Markov Chain Monte Carlo (MCMC) Method.

Future trends in AI (Quantum AI, Human AI Collaboration, Explainable AI, AI in Edge Computing, Artificial General Intelligence), Other AI applications, Natural language processing (NLP), Computer vision, Machine learning, Robotics AI in business intelligence, AI in healthcare, AI in education, AI in finance, AI in manufacturing.

45 PERIODS

LIST OF LAB EXPERIMENTS **30 PERIODS**

1. Develop a python program to simulate the agent with suitable environment to decide the numbers of papers to be purchased by observing the stock history and prize.
2. Write a python program to represent a priority queue environment and agent to support the following searching strategies.
    a. A * Search
    b. Best-first search
    c. Depth First Search.
3. Write a python Program to Implement Tic-Tac-Toe game.
4. Write a python Program to Implement 8-Puzzle problem.
5. Write a python Program to Implement Water-Jug problem
6. Write a Python Program to Implement Travelling Salesman Problem.
7. Write a python Program to Implement Tower of Hanoi.
8. Write a python Program to Implement Monkey Banana Problem.
9. Write a python Program to Implement Alpha-Beta Pruning.
10. Write a python Program to Implement 8-Queens Problem.
11. Write a python program to simulate the crossword puzzle problem with 10 words that satisfy suitable constraints in a domain.
12. Write a python program to simulate a knowledge base with a list of clauses and askables in order to make top-down inference, also creates a dictionary that maps each atoms into the set of clauses with that atom in the head.
13. Write a python program to solve the blocks world problem robot moment planning problem using the following algorithms.
    a. Forward planning
    b. Regression planning
    c. Constrain satisfaction planning
    d. Partial order planning
14. Implement Q-Learning and SARSA.
15. Implement the financial risk analysis using Monte Carlo method.

Course Outcome:

After completion of course, students would be able to:

CO1: Identify the scope for agent-based engineering solutions using AI based tools.

CO2: Demonstrate advanced search strategies.

CO3: Master the knowledge representation, reasoning and planning based AI solutions.

CO4: Establish a semantic relationship between events and actions.

CO5: Solve Complex Decision-Making problems by applying Morkovian properties.

TEXT BOOKS:

1. Artificial Intelligence – A Modern Approach, S. Russel and P. Norvig, 3rd Edition, Pearson Education, 2010.
2. Sutton, R. S., & Barto, A. G. (2018). Reinforcement learning: An introduction (2nd ed.). MIT Press.

REFERENCES:

1. Artificial Intelligence and Machine Learning, Vinod Chandra. S.S, Anand Hareendran.S., PHI Publishers, 2014.
2. Artificial Intelligence: Structures and Strategies for complex problem solving, G. Luger, 4th Edition, Pearson Education, 2002.
3. Artificial Intelligence: A new Synthesis, J. Nilsson, Elsevier Publishers, 1998.

Course Code	C++ PROGRAMMING	L	T	P	C
CS4101		3	0	2	4

COURSE OBJECTIVES:

1. To give a foundation in C programming.
2. To provide comprehensive understanding of object-oriented principles.
3. To gain advanced knowledge of the concepts such as inheritance and polymorphism in C++.
4. To equip with advanced C++ skills in exception handling and generic programming.
5. To handle the files using 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 OBJECT ORIENTED PROGRAMMING FUNDAMENTALS **12**

C++ Programming features - Data Abstraction - Encapsulation - class - object - constructors - static members – constant members – member functions – pointers – references - Role of this pointer – Storage classes – function as arguments.

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 OBJECT ORIENTED PROGRAMMING CONCEPTS **12**

String Handling – Copy Constructor - Polymorphism – compile time and run time polymorphisms – function overloading – operators overloading – dynamic memory allocation - Nested classes - Inheritance – virtual functions.

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 polymorphismto 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 totalearnings (salary + bonus for managers, hourly wage * hours worked for workers). Create instances of managers and workers and display their total earnings.

UNIT IV C++ PROGRAMMING ADVANCED FEATURES 12

Abstract class – Exception handling - Standard libraries - Generic Programming - templates – class template - function template – STL – containers – iterators – function adaptors – allocators - Parameterizing the class.
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 instances 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 mustbe thrown with a custom error message. The try block must contain the code that mightthrow an exception, and the catch block must catch the exception and display the errormessage 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 aprogram 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 loopthat allows the user to keep trying until they provide valid input. Use nested try-catchblocks to handle exceptions at different levels of the program's execution.

UNIT V FILE HANDLING CONCEPTS (Blooms Learning Levels: L3 – Apply) 12

File I/O-fstream and the File Classes-Opening and Closing a File-Reading and Writing TextFiles-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 () methodfor 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	Solve complex problems using modular and maintainable C code.
CO2	Implement object-oriented features including classes, objects, pointers and encapsulation.
CO3	Implement string handling, polymorphism and inheritance using C ++
CO4	Implement exception handling and generic programming with templates 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 HillPublishers, 2014.
2. Herbert Schildt, "C++: The Complete Reference", 5th Edition, Tata Mc-Graw HillPublishers, 2014.
3. Paul Deitel, Harvey Deitel, "C++ How to Program", 8th Edition, Prentice HallPublisher, 2016.
4. 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, Pearson Education, 2012.
3. Bjarne Stroustrup, "The C++ Programming language", 4th Edition, Pearson Education, 2013.

Course Code	DIGITAL DESIGN AND COMPUTER ORGANIZATION	L	T	P	C
EC4112		3	0	2	4

COURSE OBJECTIVES:

- To analyze and design combinational circuits.
- To analyze and design sequential circuits
- To understand the basic structure and operation of a digital computer.
- To study the design of data path unit, control unit for processor and to familiarize with the hazards.
- To understand the concept of various memories and I/O interfacing.

UNIT I COMBINATIONAL LOGIC 9

Combinational Circuits – Karnaugh Map - Analysis and Design Procedures – Binary Adder – Subtractor – Decimal Adder - Magnitude Comparator – Decoder – Encoder – Multiplexers - Demultiplexers

UNIT II SYNCHRONOUS SEQUENTIAL LOGIC 9

Introduction to Sequential Circuits – Flip-Flops – operation and excitation tables, Triggering of FF, Analysis and design of clocked sequential circuits – Design – Moore/Mealy models, state minimization, state assignment, circuit implementation - Registers – Counters.

UNIT III 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.

UNIT IV PROCESSOR 9

Instruction Execution – Building a Data Path – Designing a Control Unit – Hardwired Control, Microprogrammed Control – Pipelining – Data Hazard – Control Hazards.

UNIT V MEMORY AND I/O 9

Memory Concepts and Hierarchy – Memory Management – Cache Memories: Mapping and Replacement Techniques – Virtual Memory – DMA – I/O – Accessing I/O: Parallel and Serial Interface – Interrupt I/O – Interconnection Standards: USB, SATA

45 PERIODS

PRACTICAL EXERCISES: 30 PERIODS

1. Verification of Boolean theorems using logic gates.
2. Design and implementation of combinational circuits using gates for arbitrary functions.
3. Implementation of 4-bit binary adder/subtractor circuits.
4. Implementation of code converters.
5. Implementation of BCD adder, encoder and decoder circuits
6. Implementation of functions using Multiplexers.
7. Implementation of the synchronous counters
8. Implementation of a Universal Shift register.
9. Simulator based study of Computer Architecture

COURSE OUTCOMES:

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

CO1: Design various combinational digital circuits using logic gates
CO2: Design sequential circuits and analyze the design procedures
CO3: State the fundamentals of computer systems and analyze the execution of an instruction
CO4: Analyze different types of control design and identify hazards
CO5: Identify the characteristics of various memory systems and I/O communication

TOTAL:75 PERIODS

TEXT BOOKS:

1. M. Morris Mano, Michael D. Ciletti, "Digital Design: With an Introduction to the Verilog HDL, VHDL, and System Verilog", Sixth Edition, Pearson Education, 2018.
2. David A. Patterson, John L. Hennessy, "Computer Organization and Design, The Hardware/ Software Interface", Sixth Edition, Morgan Kaufmann/Elsevier, 2020.
3. REFERENCES
4. Carl Hamacher, Zvonko Vranesic, Safwat Zaky, Naraig Manjikian, "Computer Organization and Embedded Systems", Sixth Edition, Tata McGraw-Hill, 2012.
5. William Stallings, "Computer Organization and Architecture – Designing for Performance", Tenth Edition, Pearson Education, 2016.
6. M. Morris Mano, "Digital Logic and Computer Design", Pearson Education, 2016.

Course Code	EMPLOYABILITY ENHANCEMENT SKILLS – I	L	T	P	C
ES4101		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.

COURSE DESCRIPTION:

focuses on developing essential competencies for success in the professional world. It covers communication skills, workplace etiquette, teamwork, and collaboration principles, preparing participants for effective engagement in various work environments. Through practical exercises and interactive sessions, this program aims to enhance employability and career readiness.

Prerequisites

- Basic proficiency in communication skills.
- Familiarity with workplace etiquette and professionalism.
- Understanding of teamwork and collaboration principles.

UNIT I NUMBERS **6**

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 **6**

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 **6**

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 **6**

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 6

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

30 PERIODS

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. "Quantitative Aptitude for Competitive Examinations" by R.S. Aggarwal – 2022"
2. "Teach Yourself Quantitative Aptitude" by Arun Sharma – 2017
3. "A modern approach verbal and non – verbal reasoning" by R.S. Aggarwal – 2017

REFERENCES:

1. "Shortcuts in Mathematics" by Akhilesh Khare – 2016
2. "Vedic maths for competitive exams" by Ravi Shankar – 2016
3. "Quantitative Aptitude for Competitive Examination" by Abhijit Guha – 2017

TOTAL: 30 PERIODS

Semester II

Course Code	MATHEMATICS – II	L	T	P	C
MA4201	PROBABILITY AND STATISTICS	3	1	0	4

COURSE OBJECTIVES:

The main objectives of this course are to:

- To introduce the basic concepts of probability and distributions.
- To learn the basic concepts of two -dimensional random variables.
- To acquire the knowledge in random processes, stationary, Markov and Poisson process.
- To acquire the knowledge of testing of hypothesis for small and large samples this plays an important role in real life problems.
- To summarise the design of experiments in the field of agriculture.

UNIT I PROBABILITY AND RANDOM VARIABLES 12

Probability: Axioms – Sample space – Events - Conditional Probability – Baye's Theorem - Discrete and continuous random variables - Moments - Moment generating functions - Standard distributions: Binomial - Poisson - Geometric - Uniform - Exponential and Normal distributions.

UNIT II TWO DIMENSIONAL RANDOM VARIABLES 12

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

UNIT III RANDOM PROCESSES AND MARKOV CHAINS 12

Random Processes: Introduction and Classification -Stationary Processes -Markov Processes - Poisson Processes -Discrete Parameter Markov Chains -Chapman-Kolmogorov Equations (Statement only) -Limiting Distributions.

UNIT IV 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, F and Chi-square distributions for mean, variance **-** Contingency table (test for independent) - Goodness of fit.

UNIT V DESIGN OF EXPERIMENTS 12

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

TOTAL: 60 PERIODS

COURSE OUTCOMES:

Upon completion of the 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: Recognize the basic concepts of two - dimensional random variables and apply in engineering applications.

CO3: Develop the basic concepts of random processes which are widely used in engineering fields.

CO4: Apply the concept of testing of hypothesis for small and large samples in real life problems.

CO5: Investigate of design of experiments in the field of agriculture.

TEXT BOOKS:

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. Ibe, O.C.," Fundamentals of Applied Probability and Random Processes ", 1st Indian Reprint, Elsevier, 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. 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.

Course Code	TAMILS AND TECHNOLOGY	L	T	P	C
HS4201		1	0	0	1

COURSE OBJECTIVE:

The main objectives of this course are to:

- Investigate weaving and ceramic techniques during the Sangam Age.
- Examine design, construction, and architectural marvels of ancient Tamil Nadu.
- Explore ancient manufacturing methods and industries through archaeological evidence.
- Analyze agricultural and irrigation practices during historical periods.
- Study the evolution of Scientific Tamil and Tamil computing.

UNIT I WEAVING AND CERAMIC TECHNOLOGY 3

Weaving Industry during Sangam Age – Ceramic technology – Black and Red Ware Potteries (BRW) – Graffiti on Potteries.

UNIT II DESIGN AND CONSTRUCTION TECHNOLOGY 3

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 Husbandr – 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.

15 PERIODS

TEXT 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 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 and Educational Services Corporation, Tamil Nadu)

REFERENCE:

1. Journey of Civilization Indus to Vaigai (R. Balakrishnan) (Published by: RMRL)

TOTAL: 15 PERIODS

Course Code	COMMUNICATIVE ENGLISH –II	L	T	P	C
HS4202		3	0	2	4

COURSE OBJECTIVES

The main objectives of this course are to:

- To engage learners in meaningful language activities to improve their LSRW skills
- To enhance learners' awareness of general rules of writing for specific audiences
- To help learners understand the purpose, audience, and contexts of different types of writing
- To develop analytical thinking skills for problem-solving in communicative contexts
- To cultivate the learners to understand job applications and interviews for internships and placements.

UNIT I MAKING COMPARISONS 9

Listening – Evaluative Listening: Advertisements, Product Descriptions, – Audio/ video Speaking– Marketing a product, Persuasive Speech Techniques.

Reading – Reading advertisements, user manuals, and brochures Writing – Letter to the editor; Compare and Contrast Essay Grammar – Impersonal passive voice; Prepositional phrases Vocabulary –Contextual meaning of words

UNIT II EXPRESSING CASUAL RELATIONS IN SPEAKING AND WRITING 9

Listening – Listening to longer technical talks and completing–gap-filling exercises. Listening to 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.

UNIT III PROBLEM SOLVING 9

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.

UNIT IV REPORTING OF EVENTS AND RESEARCH 9

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.

UNIT V THE ABILITY TO PUT IDEAS OR INFORMATION COGENTLY 9

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.

45 PERIODS

COURSE OUTCOMES

After the completion of the course, 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.

LIST OF EXERCISES:

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

30 PERIODS

TEXT BOOKS:

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	WEB DEVELOPMENT ESSENTIALS	L	T	P	C
CS4201		3	0	2	4

COURSE OBJECTIVES:

The main objectives of this course are to:

- To understand the web programming using Hypertext Markup Language (HTML)
- To develop responsive websites using Cascading Style Sheets (CSS)
- To implement interactive and dynamic effects on web pages using JavaScript.
- To develop dynamic web applications using Document Object Model (DOM).
- To develop dynamic web applications with Asynchronous JavaScript and deployment in a GitHub repository.

Prerequisites

- Proficiency in at least one programming language (e.g., Python, Java, JavaScript).
- Understanding of software development fundamentals (e.g., version control, debugging).
- Familiarity with basic data structures and algorithms.

UNIT I HYPERTEXT MARKUP LANGUAGE (HTML) 15

Introduction to HTML – HTML Document Structure –Text – Lists -Frames, Table and Div tags – 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) 15

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. SASS.

Coding Exercises

2. Design a web page using HTML and CSS.
3. 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 properties for background, text effects, positioning, link).

UNIT III JAVASCRIPT **15**

Introduction to JavaScript – Variables - Data Types – Operators - Statements and Expressions- Strings and Template Literals - if / else Statements - switch Statements - Functions – Arrays – Classes and Objects-Looping Statements – CSS in JS (Styled-Components/emotions), Ecmascript, Closures, Polyfills

Coding Exercises

1. Write a java script program to Check if a Number is Odd or Even.
2. Write a java script 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) **15**

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- Shadow DOM.

Coding Exercises

1. Write a java script program to find and change the content of the HTML elements.
2. Write a java script program to handle events (Mouse/Keyboard/Form events).

Problems-solving Assignments

1. Write a java script program to perform replacement and removal of HTML elements.

UNIT V ASYNCHRONOUS JAVASCRIPT AND GITHUB **15**

JavaScript: Behind the scene - JS Callbacks - JS Asynchronous - JS Promises - JS Async/Await– Git - GIT environment and setting up - GIT commands- GitHub – Pushing files to GitHub.

Coding Exercises

1. Write a java script program to display a text using set Timeout method.
2. Write a program to perform java script callback function.

Problems-solving Assignments

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

75 PERIODS

COURSE OUTCOMES:

Upon completion of the 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 java script and deploy the webpage using Github repository

TEXT BOOKS:

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 Responsive Websites 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)

REFERENCE BOOKS:

1. P.J. Deitel, H.M. Deitel, and Nieto, "Internet and World Wide Web - How to Program", Prentice Hall, 6th Edition, 2020.
2. Mark Myers, "A Smarter Way to Learn Javascript: The New Approach That Uses Technology to Cut Your Effort in Half "Kindle edition,2014.
3. "Learning JavaScript: Add Sparkle and Life to Your Web Pages" by Ethan Brown (3rd Edition, 2021)
4. "JavaScript Everywhere: Building Cross-Platform Applications with GraphQL, React, React Native, and Electron" by Adam D. Scott (2020)

Course Code	DATA STRUCTURES USING C++	L	T	P	C
CS4202		3	0	2	4

COURSE OBJECTIVES:

The main objectives of this course are to:

- To understand the basic concepts of linear data structures like arrays and linked lists.
- To grasp the concept of stacks and queues as a linear data structure and the operations upon them.
- To understand the fundamental concepts of hierarchical Tree data structures.
- To explore optimization strategies for indexing structures and graph algorithms.
- To understand the concept of searching for quick data retrieval, sorting for arranging data, hash functions strategies for optimized data storage.

UNIT I LISTS **15**

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.

Coding Exercises:

1. Write a C++ programs to implement single linked list ADT to perform following operations.
2. insert an element into a list.
3. delete an element from list.
4. search for a key element in list.
5. count number of nodes in list.
6. Write a C++ programs to implement doubly linked list ADT to perform following operations.
7. insert an element into a list at the end.
8. delete middle element from list.
9. print the element in reverse order.
10. Write a C++ programs to Perform Polynomial Addition using singly linked list.

Problems-solving Assignments:

1. Write a C++ program to find the sum of all elements in an array.
2. Write a C++ program that implements and demonstrates a circularly linked list operations such as insertion, deletion, and traversal.
3. Write a C++ program that demonstrates the Radix Sort algorithm using linked list.

UNIT II STACKS AND QUEUES **15**

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

Coding Exercises:

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

Problems-solving Assignments:

1. Write a C++ program to implement a circular queue operations enqueue, dequeue, front, isEmpty using a linked list
2. Write a C++ program to check if a given expression with parentheses is balanced using a stack.
3. Write a C++ program to implement a Deque (double-ended queue) operations pushFront(), pushBack(), popFront(), popBack(), front(), back(), and isEmpty() using a doubly linked list.

UNIT III TREES **15**

Tree ADT – Tree Traversals – Binary Tree ADT – Expression trees – Binary Search Tree ADT – AVL Trees – Splay trees-Priority Queue (Heaps) – Binary Heap.

Coding Exercises:

1. Write a C++ program to implement a binary search tree with the following operations:
   a) Insert an element into a binary search tree.
   b) Delete an element from a binary search tree.
   c) Search for a key element in a binary search tree.
2. Write a C++ program that uses 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 implement 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 **15**

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's algorithm.

Coding Exercises:

1. Write a C++ program that demonstrates B-Tree operation insertion, search, and display.
2. Write a C++ program for Dijkstra's single source shortest path algorithm.
3. Write a C++ program for Prim's MinimumSpanning Tree (MST) algorithm.

Problems-solving Assignments:

1. Write a C++ program that demonstrates Breadth-First Traversal (BFS) in a graph using 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 Minimum Spanning Tree (MST) of a graph.

UNIT V SEARCHING, SORTING AND HASHING TECHNIQUES 15

Searching – Serial Search – Binary Search. Sorting – Insertion sort –Merge Sort-Quick sort-Heap sort - Hashing– Hash Functions – Separate Chaining –Hash table without linked list– Rehashing –Extendible Hashing-A Packaging System for C++

Coding Exercises:

1. Write a C++ program 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.

75 PERIODS

COURSE OUTCOMES:

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

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

TEXT BOOKS:

1. Langsam, Augenstein, Tenenbaum, “Data Structures using C & C++”, 2nd Edition, Pearson Education, 2015.
2. Mark Allen Weiss, “Data Structures and Algorithm Analysis in C++”, 3rd Edition, Addison Wesley Publishing Company, 2006.
3. Sartaj Sahni, “Data Structures, Algorithms, and Applications in C++”, 2nd Edition, Universities Press, 2005.

REFERENCES:

1. MT Goodrich, “Data Structures and Algorithms in C++”, 2nd Edition, John Wiley & Sons Inc,2011.
2. Brijendra Joshi, “Data Structures and Algorithms in C++”, 1st Edition, McGraw Hill Education, 2010.
3. Sartaj Sahni, Ellis Horowitz, “Fundamentals of Data Structures in C++”, 2nd Edition, Universities Press, 2008.

Course Code	DATABASE MANAGEMENT SYSTEMS	L	T	P	C
CS4203		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 a 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:

1. 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:
   (a) courses, including code, title, credits, syllabus, and prerequisites;
   (b) course offerings, including course number, year, semester, section number, instructor(s), timings, and classroom;
   (c) students, including student-id, name, and program;
   (d) 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.
3. Construct an E-R diagram for the registrar's office. Document all assumptions that you make about the mapping constraints.

UNIT II RELATIONAL MODEL 12

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, Savepoint, 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

Actor	Ayear
Cage	1964
Hanks	1956
Maguire	1975
McDormand	1957

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

Write the following relational algebra queries for a given set of relations.
Find movies made after 1997
Find movies made by Hanson after 1997
Find all movies and their ratings
Find all actors and directors
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,No,Startdate)
Perform the following:
Create company database.
Viewing all databases.
Viewing all Tables in a Database.
Creating Tables (With and Without Constraints).
Inserting/Updating/Deleting Records in a Table. 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 employee.
   5. Retrieve distinct number of 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)
   1. Display the resulting salaries if every employee working on the 'Research' Department is given a 10 percent raise.
   2. 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.
   3. Retrieve the names of employees Controlled by department number 5 (use EXISTS operator).
   4. Retrieve the name of each dept and number of employees working in each department which has at least 2 employees.
   5. Retrieve the names of employees who were born in the 1990's. Retrieve the names of employees and their dept name (using JOIN)

6. **Perform the String Functions, Date functions and Mathematical functions.**

7. **For the given tables:**
   EMPLOYEE(Ename, MNIT, LNAME ,No, EDate, Address, gender, Salary, Dno)Perform the Following
   1. Creating Views (With and Without Check Option),
   2. Selecting from a View
   3. Dropping Views,

8. **Write a Pl/SQL program to print integers from 1 to 10 by using PL/SQL FOR loop.**
9. **Given the table:**
   EMPLOYEE (EmpNo, Name, Salary, Designation, DeptID)
   Write a cursor to select the five highest paid employees from the table.

Given an integer i, write a PL/SQL procedure to insert the tuple (i, 'xxx') into a given relation.

Create a trigger to insert the updated salary details in employee table.

Write the SQL queries to implement the following concepts:

Constraints of all types with more schemas.

Create materialized View Procedure with single schema. Problem-solving Assignments:

EMPLOYEE TABLE with CHECK CONSTRAINT for

CITY:employee (empid, empname, st, city, phone, pin)

WORKTABLE with DEFAULT CONSTRAINT for COMPANYNAME:

worktable (empid, companyname, salary) COMPANY TABLE: -

company(companyna me, city)MANAGER TABLE: - manager(empid, managername)

USE REFERENCE INTEGRITY for WORKTABLE and EMPLOYEE tables.

Write the SQL Queries for the following

Find name, city, phone, pin of the resident of all employees who work for TCS

Find all employee names who don't work for TCS

Find all employees who lives in particular city.

Find the employee who got highest salary company wise.

Find the employee id whose salary is greater than 1 lakh

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 it 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 have 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 –

Rename the column product_price of product relation to new_product_rate.

Display the order number and date on which the clients placed their order.

Delete all the records having delivery date before 14thOctober 2015.

Find the sum of products based on manufacturer wise.

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 FifthNormal Form.

Problems-solving Assignments:

1. Design a database schema for a hospital management system. The system stores information about patients, doctors, medical records, and appointments.
Patients (PatientID, FirstName, LastName, Birthdate, Address, Phone) Doctors (DoctorID, FirstName, LastName, Specialty) Medical Records (RecordID, PatientID, DoctorID, Diagnosis, Prescription) Appointments (AppointmentID, PatientID, DoctorID, Appointment Date)
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:

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

12

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:

"RAID mechanism improves reliability through redundancy". Justify this statement. Also, discuss the best situations to be adapted for the different levels of RAID?

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):

Database design and implementation of Pay roll Processing.
Database design and implementation of Banking System.
Database design and implementation of Movie Ticket Booking System.
Database design and implementation of Hospital Management System.
Database design and implementation of Job Recruitment Portal.
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.

TEXT BOOKS:

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", FourthEdition, Pearson Addison 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
CS4204		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 a 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.

Functional Programming in Java: Introduction to Functional Programming Concepts – Lamda Expressions – Functional Interfaces – Streams API.

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. The program 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 inherited classes 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, Triangle and 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.

Features of Java 21: Overview of new features – Virtual Threads – Pattern Matching of switch – Record Patterns.

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. The program 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:

2. 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 inherited classes 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.

3. 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, Triangle and 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 Character streams – Reading and Writing Console – Reading and Writing Files.

Enhancements of Java 21: Foreign Function & Memory API improvements – Deprecated and Removed APIs – New string methods.

Coding Exercises:

1. Write a JAVA program divides by zero to cause an Arithmetic Exception. 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 Custom Exception that inherits from the standard Exception class. This custom exception class has a constructor that takes a message as an argument 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 the stream 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.
Advanced Multithreading (Java 21): Introduction to Virtual Threads - Structured Concurrency.

Coding Exercises:

1. Write a Java program that creates three threads. First thread displays “Good Morning” every one second, 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 computes the square of the number and prints. If the value is odd, the third thread will print the value of the cube of 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 Class LinkedList 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.
Enhancements in Collection Framework – Improvements in Pattern Matching – Record Patterns.

Coding Exercises:

1. Create a java program for getting a list of fruits. Print the list using for loop. Add one fruit to the 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 write the 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 JDBC Demo. 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.

TOTAL NUMBER OF PERIODS: 60

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 concept’s 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

TEXT BOOKS:

1. Herbert Schildt, “Java: The Complete Reference”, 12th Edition, McGraw Hill Education, Ne Delhi, 2021.
2. "Core Java Volume I – Fundamentals" by Cay S. Horstmann, 12th Edition, 2022.
3. "Core Java Volume II – Advanced Features" by Cay S. Horstmann, 12th Edition, 2022.

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
ES4201		0	0	2	2

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 **5**

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 **8**

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 **5**

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 **7**

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 **5**

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

30 PERIODS

COURSE OUTCOMES

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

At the end 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. Quantitative Aptitude for Competitive Examinations by R.S. Aggarwal – 202
2. Teach Yourself Quantitative Aptitude by Arun Sharma – 2017
3. A modern approach to verbal and non-verbal reasoning by R.S. Aggarwal – 2017

REFERENCES:

1. Shortcuts in Mathematics by Akhilesh Khare – 2016
2. Vedic maths for competitive exams by Ravi Shankar – 2016
3. Quantitative Aptitude for Competitive Examination by Abhijit Guha – 2017

Course Code	NCC CREDIT COURSE	L	T	P	C
AD42XX		2	0	0	2

UNIT I NCC General **6**

Aims, Objectives & Organization of NCC (1) – Incentives (2) – Duties of NCC Cadet (1) – NCC Camps: Types & Conduct (2)

UNIT II National Integration and Awareness **4**

National Integration: Importance & Necessity (1) – Factors Affecting National Integration (1) – Unity in Diversity & Role of NCC in Nation Building (1) –Threats to National Security (1)

UNIT III Personality Development **7**

Self-Awareness, Empathy, Critical & Creative Thinking, Decision Making and Problem Solving (2) – Communication Skills (3) – Group Discussion: Stress & Emotions (2)

UNIT IV Leadership **5**

Leadership Capsule: Traits, Indicators, Motivation, Moral Values, Honour Code (3) – Case Studies: Shivaji, Jhansi Ki Rani (2)

UNIT V Social Service and Community Development **8**

Basics, Rural Development Programmes, NGOs, Contribution of Youth (3) – Protection of Children and Women Safety (1) Road/ Rail Travel Safety (1) – New Initiatives (2) – Cyber and Mobile Security Awareness (1)

TOTAL: 30 PERIODS

Levels:

Level	**Activities**
1 ARMY	II semester: The NCC cadets are fresh to NCC activities are provided with above curriculum / syllabus
2 NAVY	For the Higher semester cadets of IV semester the credits are awarded with the basic qualifying needs Minimum one Combined Annual training Camp (CATC) or Attachment Camp or Centrally Organised Camp has to be attended (camp certificate is required) Appeared for B Certificate Exam
3 AIR FORCE	For the Higher semester cadets of VI semester the credits are awarded with the basic qualifying needs Minimum one Combined Annual training Camp (CATC) or Attachment Camp or Centrally Organised Camp has to be attended (camp certificate is required) Appeared for C Certificate Exam

Semester III

Course Code	DISCRETE MATHEMATICS	L	T	P	C
MA4302		3	1	0	4

COURSE OBJECTIVES:

The main objectives of this course are:

1. To familiarize the applications of algebraic structures.
2. To understand the concepts of Permutations, Combinations and Induction.
3. To acquire the knowledge of graph models.
4. To extend student's logical and mathematical maturity and ability to deal with abstraction.
5. To understand the concepts, significance of lattices and Boolean algebra which are widely used in computer science and engineering.

UNIT I SETS AND ALGEBRAIC STRUCTURES 12

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

UNIT II COMBINATORICS 12

Introduction to Basic Counting Principles - Permutations and combinations - simple problems - Pigeonhole Principle - Recurrence relations - Generating Functions - Introduction to Proof Techniques - Mathematical Induction.

UNIT III GRAPH THEORY 12

Graphs: Definition & basic properties - adjacency and incidence matrix - Isomorphism - Eulerian & Hamiltonian Paths & Circuits - Connectivity and Reachability -Trees- Matching & Colouring: Basics concepts of graph matching, graph colouring problem - Independence number and clique number, chromatic number, statement of Four-colour theorem - Planar graphs, Euler's formula, dual of a planar graph.

UNIT IV PROPOSITIONAL LOGIC 12

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.

UNIT V LATTICES AND BOOLEAN ALGEBRA 12

Partial ordering – Posets – Lattices as posets – Properties of lattices – Lattices as algebraic systems – Sub lattices – Direct product and homomorphism – Some special lattices – Introduction of Boolean algebra, truth table, basic logic gate, basic postulates of Boolean algebra, principle of duality, canonical form, Karnaugh map.

TOTAL: 60 PERIODS

COURSE OUTCOMES:

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

CO1: Understand the algebraic structures on many levels.
CO2: Learn the concepts of combinatorics and its applications in real life problems.

CO3: Understand the concepts of graph theory, natural applications of trees and apply the graph colouring concepts in partitioning problems.
CO4: Illustrate the concept of logics, propositional calculus and normal form.
CO5: Apply the ideas of lattices and Boolean algebra in solving computer programming problems.

TEXT BOOKS:

1. J.P. Tremblay, R. Manohar. "Discrete Mathematical Structures with Applications to Computer Science", McGraw-Hill Education, 2019.
2. Kenneth H. Rosen. "Discrete Mathematics and Its Applications", 8th Edition, McGraw-Hill Education, 2021.
3. J. A. Bondy and U. S. R. Murty, "Graph Theory with Applications", Macmillan Press, London, 1976.

REFERENCES:

4. Ralph P. Grimaldi, "Discrete and Combinatorial Mathematics", 5th Edition, Pearson Education, 2019.
5. Marc Lipson, Seymour Lipschutz. "Discrete Mathematics (Schaum's Outlines)", 4rd Edition, McGraw-Hill Education, 2022.
6. Thomas Koshy, "Discrete Mathematics with Applications", Elsevier India, 2005.
7. L. Zhongwan. "Mathematical Logic for Computer Science", World Scientific, Singapore,1989.
8. N. Deo. "Graph Theory with Applications to Engineering and Computer Science", Prentice Hall, 1979.

Course Code	OPERATING SYSTEM	L	T	P	C
AD4301		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 Need of Virtual machines- OS design considerations for multiprocessor and multicore - Operating System generation - System boot.

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 –Multi core programming - 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 SECURITY, VIRTUAL MACHINES AND MOBILE OS **9**

UNIX Security – UNIX Protection System – UNIX Authorization – UNIX Security Analysis – UNIX Vulnerabilities - Windows Vulnerabilities – Address Space Layout Randomizations – Retrofitting Security into a Commercial Operating System – Introduction to Security Kernels-Virtual Machines – History, Benefits and Features, Building Blocks, Types of Virtual Machines and their 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 a Deadlock Detection Algorithm
9. Write a C program to implement Threading
10. Implement the paging Technique using the 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

TEXT BOOKS:

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”, Tata McGraw Hill Edition, 2010.
2. William Stallings, "Operating Systems: Internals and Design Principles", 7th Edition, PrenticeHall, 2018.
3. Achyut S. Godbole, Atul Kahate, “Operating Systems”, McGraw Hill Education, 2016.

Course Code	DESIGN AND ANALYSIS OF ALGORITHMS	L	T	P	C
AD4302		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 the graph algorithm.
3. To understand different algorithm design techniques
4. To solve programming problems using a state space tree
5. To understand the concepts behind NP-completeness, Approximation algorithms, and randomization 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. Sorting: Insertion sort – heap sort.

UNIT II GRAPH TECHNIQUE **9**

Graph algorithms: Representations of graphs - Graph traversal: DFS – BFS - applications - Connectivity, strong connectivity, bi-connectivity - Minimum spanning tree: Kruskal's andPrim's algorithm- Shortest path: Bellman-Ford algorithm - Dijkstra's algorithm - Floyd- Marshall algorithm - Huffman coding - Network flow: Flow networks - Ford-Fulkerson method – Matching: 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. - Activity-selection problem –- Optimal Merge pattern-0/1 knapsack problem.

UNIT IV STATE SPACE SEARCH BACKTRACKING **9**

Backtracking: n-Queens problem - Hamiltonian Circuit Problem - Subset Sum Problem – Graph coloring 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 Heap Sort.

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 like pattern 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:
   a) Explain the Divide and Conquer approach for finding the maximum and minimum in an array.
   b) Analyze the time complexity of the Divide and Conquer solution.
   c) Discuss any limitations or scenarios where this approach might be preferable.
2. Matrix-Chain Multiplication:
   a) Describe the Dynamic Programming approach for solving the Matrix-Chain Multiplication problem.
   b) Analyze the time and space complexity of the Dynamic Programmingsolution.
   c) Discuss the advantages of using Dynamic Programming in this context.
3. Comparison:
   a) Compare the Divide and Conquer and Dynamic Programming approachesin terms of time and space complexity.
   b) Discuss scenarios where one approach might be more suitable than the other.

COURSE OUTCOMES:

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

CO1: Gain proficiency in understanding and applying algorithm analysis techniques to searching and sorting algorithms.
CO2: Develop the ability to critically analyze the efficiency of graph algorithms.
CO3: Acquire a comprehensive understanding of various algorithm design techniques.
CO4: Enhance problem-solving skills by utilizing state space trees for programming challenges.
CO5: Understand the concepts behind NP-completeness, approximation algorithms, and randomized algorithms.

TEXT BOOKS:

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, Pearson Education, 2012.
2. Alfred V. Aho, John E. Hopcroft and Jeffrey D. Ullman, "Data Structures and Algorithms", Reprint Edition, Pearson Education, 2006.
3. S. Sridhar, “Design and Analysis of Algorithms”, Oxford University Press, 2014.

Course Code	WEB DEVELOPMENT FRAMEWORKS AND PRACTICES	L	T	P	C
AD4303		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 a 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 intoMongoDB – 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

Project – Simple weather application using MongoDB
Project – URL shortener Application using SQL.
Project - Flight Ticket Booking

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

Type of Users

User
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 Develop business logic for back end using Node.js.
CO2 Create full stack application using Express.js.
CO3 Develop Python based full stack applications.
CO4 Develop strong querying and analytics using MongoDB.
CO5 Develop robust java applications using Spring boot and hibernate.

TOTAL: 75 PERIODS

TEXT BOOKS:

1. Jonathan Wexler, “Get Programming with Node.js”, Manning Publications, 2019.
2. Beginning Node.js, Express & MongoDB Development, Greg Lim, 1st Edition, 2019.
3. "Node.js Design Patterns: Design and implement production-grade Node.js applications using proven patterns and techniques" by Mario Casciaro, Luciano Mammino (3rd Edition, 2020)
4. "MongoDB: The Definitive Guide: Powerful and Scalable Data Storage" by Shannon Bradshaw, Eoin Brazil, Kristina Chodorow (3rd Edition, 2019).
5. "Spring in Action" by Craig Walls (6th Edition, 2022)

ONLINE RESOURCES:

1. https://nodejs.org/docs/latest/api/
2. https://expressjs.com/
3. https://www.udemy.com/course/the-complete-nodejs-developer-course-2/
4. https://www.fullstackpython.com/
5. https://learn.mongodb.com/learning-paths/introduction-to-mongodb
6. https://spring.io/projects/spring-framework#learn

Course Code	DATA SCIENCE AND VISUALIZATION	L	T	P	C
AD4304		3	0	2	4

COURSE OBJECTIVES

1. Explore data science roles, responsibilities, and problem-solving capabilities.
2. Master R programming for data visualization and analysis.
3. Develop proficiency in Tableau for creating diverse visualizations.
4. Utilize advanced Tableau features for in-depth data analysis and dashboard creation.
5. Learn effective data preparation techniques and storytelling with Tableau.

UNIT I INTRODUCTION TO DATA SCIENCE 9

Data Science Roles: A Brief History, Data Science – Engineering – Analysis - Modeling/Inference, What Kind of Questions Can Data Science Solve? – Prerequisites - Problem Type, Structure of Data Science Team, Data Science Roles.

Soft Skill for Data Science: Statistician vs Data Scientist, Beyond Data and Analytics, Three Pillars of Knowledge, Data Science Project Cycle - Types of Data Science Projects - Problem Formulation and Project Planning Stage -Project Modeling Stage - Model Implementation and Post Production Stage - Project Cycle Summary, Common Mistakes in Data Science - Problem Formulation Stage - Project Planning Stage - Project Modeling Stage - Model Implementation and Post Production Stage - Summary of Common Mistakes

UNIT II R PROGRAMMING FOR DATA SCIENCE 9

Explore – Data Visualization with ggplot2 – Data Transformation with dplyr – Exploratory Data Analysis, Wrangle – Tibble with tibble – Data Import with readr – Tidy Data with tidyr – Relational Data with dplyr – Strings with stringr – Factors with forcats – Dates and Times with lubridate, Program – Pipes with magrittr – Functions – Vectors – Iteration with purr, Model – Model basics with modelr – Model Building – Many Models with Purrr and broom, Communicate – R Markdown – Graphics for Communication with ggplot2 – R Markdown Formats.

UNIT III DATA VISUALIZATION USING TABLEAU FUNDAMENTALS 9

Connecting to Data, Foundations for building visualizations, visualizing data – bar charts - line charts - Geographical visualizations - show me, File types, Joins and blends, Filtering data, Comparing values – bullet chart – Bar-in-Bar chart – Highlighting categories of Interest, Visualizing Dates and Times – Gantt Charts, Stacked bars, Tree maps, Area Charts, Pie Charts, Visualizing Distributions.

UNIT IV LEVERAGING THE FULL POWER OF TABLEAU 9

Four types of calculations – Row level – Aggregate level – Level of Details (LoD – Fixed, Include, Exclude) – Parameters – Ad hoc Calculations, Table calculations – Relative vs Fixed – Custom table calculations, Leverage Formatting – Workbook level – worksheet Level – Field Level, Tooltips, Dashboards – Definition – Objectives – Approaches – Designing – Designing for different displays and devices – How Action works, Trends.

UNIT V DATA PREPARATION AND ADVANCED TECHNIQUES 9

Cleaning and Structuring Messy Data – Structuring Data for Tableau – Dealing with data Structures, Tableau Prep - Connecting to data - cleaning data - Filtering Tableau Prep - Transforming data for Analyst – Options for Automatic flows, Story Telling – Creation – Presenting -Printing – Exporting – Sharing with Tableau Desktop, Tableau Reader, Tableau Server, Tableau Online, Tableau Public.

45 PERIODS

LIST OF EXPERIMENTS 30 PERIODS

1. Utilize Tableau to visualize global temperature anomalies over time using the NASA GISS Surface Temperature Analysis (GISTEMP) dataset. (Dataset Link: https://data.giss.nasa.gov/gistemp/)
2. Explore the World Bank World Development Indicators dataset in Tableau to visualize global trends in economic indicators such as GDP growth, poverty rates, and education spending. (Dataset Link: https://data.worldbank.org/indicator)
3. Analyze the U.S. Energy Information Administration (EIA) dataset using Tableau to visualize trends in energy production, consumption, and renewable energy usage in the United States. (Dataset Link: https://www.eia.gov/opendata/)
4. Utilize Tableau to visualize global COVID-19 vaccination rates, distribution efforts, and vaccine efficacy using data from Our World in Data. (Dataset Link: https://ourworldindata.org/covid-vaccinations)
5. Explore the Kaggle Customer Churn dataset in Tableau to visualize customer churn trends, factors influencing churn, and retention strategies for businesses. (Dataset Link: https://www.kaggle.com/blastchar/telco-customer-churn)
6. Analyze the Spotify Million Playlist Dataset using Tableau to visualize music listening trends, popular genres, and song characteristics across different regions. (Dataset Link: https://www.aicrowd.com/challenges/spotify-million-playlist-dataset-challenge)
7. Utilize Tableau to visualize trends in air quality, pollution levels, and environmental factors using data from the Environmental Protection Agency (EPA). (Dataset Link: https://www.epa.gov/outdoor-air-quality-data)
8. Explore the United Nations Refugee Agency (UNHCR) dataset in Tableau to visualize global refugee populations, displacement trends, and regional refugee crises. (Dataset Link: https://www.unhcr.org/en-us/figures-at-a-glance.html)
9. Analyze the European Centre for Disease Prevention and Control (ECDC) dataset using Tableau to visualize trends in infectious diseases, vaccination rates, and

epidemiological data in Europe. (Dataset Link: https://www.ecdc.europa.eu/en/publications-data/download-todays-data-geographic-distribution-covid-19-cases-worldwide)

10. Utilize Tableau to visualize trends in e-commerce sales, consumer behavior, and online shopping preferences using the Amazon Customer Reviews dataset. (Dataset Link: https://registry.opendata.aws/amazon-reviews/)
11. Explore the Google Trends dataset in Tableau to visualize search interest over time for various topics, keywords, and products on Google's search engine. (Dataset Link: https://trends.google.com/trends/explore)
12. Analyze the Bureau of Labor Statistics (BLS) dataset using Tableau to visualize employment trends, labor force participation rates, and job market dynamics in the United States. (Dataset Link: https://www.bls.gov/data/)
13. Utilize Tableau to visualize trends in global internet usage, digital adoption rates, and online behavior using data from the World Internet Project (WIP). (Dataset Link: https://www.worldinternetproject.net/)
14. Explore the Global Terrorism Database (GTD) in Tableau to visualize trends in terrorist incidents, attack methods, and hotspots around the world. (Dataset Link: https://www.start.umd.edu/gtd/)
15. Analyze the World Happiness Report dataset using Tableau to visualize global happiness scores, factors influencing happiness, and regional variations in well-being. (Dataset Link: https://worldhappiness.report/ed/2021/)

TOTAL PERIODS: 75

COURSE OUTCOMES

CO1: Develop a solid understanding of data science principles to optimize data storage, retrieval, and analysis for enhanced business intelligence.

CO2: Gain proficiency in utilizing R programming to uncover valuable patterns and insights from large datasets, enabling informed decision-making.

CO3: Acquire the ability to effectively categorize and predict data instances using Tableau's data visualization techniques, leading to personalized recommendations and improved risk assessment.

CO4: Master advanced Tableau features to identify natural groupings within data and detect anomalies, facilitating targeted marketing strategies and anomaly detection.

CO5: Learn efficient data preparation techniques to extract meaningful information from diverse data types, enabling comprehensive analysis and deep understanding.

TEXT BOOKS:

1. Hui Lin and Ming Li, "Practitioner’s Guide to Data Science", First Edition, 2019, TechData Publishers.
2. Hadley Wickham and Garrett Grolemund, "R for Data Science", First Edition, 2017, O'Reilly Media.

3. 1. Joshua N. Milligan, "Learning Tableau 2019, Third Edition", Third Edition, 2019, Packt Publishing.

REFERENCES:

1. Foster Provost and Tom Fawcett, "Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking", Second Edition, 2019, O'Reilly Media.
2. Paul Teetor, "R Cookbook: Proven Recipes for Data Analysis, Statistics, and Graphics", Second Edition, 2019, O'Reilly Media.
3. Daniel G. Murray, "Tableau Your Data! Fast and Easy Visual Analysis with Tableau Software", Second Edition, 2016, Wiley.
4. Joel Grus, "Data Science from Scratch: First Principles with Python", First Edition, 2015, O'Reilly Media.
5. Hadley Wickham, "Advanced R", First Edition, 2014, CRC Press.
6. Joshua N. Milligan, "Tableau Desktop: A Practical Guide for Business Users", First Edition, 2019, Packt Publishing.

Course Code	PARALLEL PROGRAMMING THROUGH PYTHON	L	T	P	C
AD4305		3	0	2	3

COURSE OBJECTIVES:

- To understand the basic concepts of python for parallel Computing
- To explore Multithreading and Multiprocessing in python
- To review the distributed computing concepts in python
- To analyze the parallel computing patterns in python
- To experiment the Parallelism in Python without libraries

COURSE DESCRIPTION:

This course is designed to provide students with the knowledge and skills necessary to leverage Python for parallel programming, enabling them to harness the power of multicore processors, clusters, and parallel programing environments effectively.

PREREQUISITES:

- Proficiency in Python programming language
- Understanding of basic computer science concepts (e.g., data structures, algorithms)
- Familiarity with fundamental concepts of parallel and concurrent programming

Unit I Python Basics for Parallel Computing **6**

Overview of the Python programming language- Python syntax essentials for high-performance computing- Setting up the Python environment for parallel computing- Introduction to Python's scientific computing stack (NumPy, SciPy)

Unit II Multi-Threading and Multi-Processing in Python **6**

Understanding Python's Global Interpreter Lock (GIL)-Creating and managing threads using the threading module-Using the multiprocessing module for CPU-bound tasks-Synchronization primitives, shared memory, and managing state

Unit III Distributed Computing in Python **6**

Introduction to distributed computing with Python- Setting up and managing worker processes with multiprocessing and concurrent futures- Using message passing with Celery for distributed tasks - Introduction to Dask for scalable analytics and parallel computing

Unit IV Parallel Computing Patterns in Python **6**

Common parallel computing patterns: Map Reduce, scatter-gather-Implementing parallel map with concurrent futures- Parallel loops, aggregations, and reductions with Dask-Task scheduling and load balancing strategies

Unit V Implementation of Parallelism in Python without Libraries **6**

Python's multiprocessing module- Python's threading module- Python's asyncio module- POSIX threads- extensions in C or Cython to leverage low-level parallelism features - integration of C or Cython into Python code.

Theory: 30 Periods

Practical Exercises

1. Write a Python script that calculates the square of numbers from 1 to 10 using multiprocessing.
2. Extend the script to utilize a multiprocessing pool to parallelize the task.
3. Create a Python program to download multiple images from URLs using multithreading.
4. Explore the Global Interpreter Lock (GIL) limitations and discuss how multithreading may or may not improve performance.
5. Implement parallel versions of common sorting algorithms such as merge sort or quicksort using multiprocessing.
6. Write a Python program to perform matrix multiplication using both serial and parallel approaches. Experiment with different matrix sizes and compare the performance of the parallel version with the serial version.
7. Use a library like Dask or PySpark to perform distributed computing tasks. Design a task such as distributed word counting or distributed matrix operations and execute it on a cluster.
8. Explore GPU parallelization using libraries like PyCUDA or Numba. Implement a computationally intensive task such as image processing or scientific computing using GPU parallelization and compare its performance with CPU-based implementations.

Practical: 30 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:
Upon completion of this course, the students will be able to:

CO1: Understand the basic concepts of Python for Parallel Computing
CO2: Explore Multithreading and Multiprocessing in Python
CO3: Analyze the distributed computing concepts in Python
CO4: Apply the parallel computing patterns in Python
CO5: Experiment and evaluate the Parallelism in Python without libraries

TEXT BOOKS:

1. Parallel Programming with Python" by Jan Palach
2. "Python Parallel Programming Cookbook" by Giancarlo Zaccone:

REFERENCES:

1. High-Performance Python: Practical Performant Programming for Humans" by Micha Gorelick and Ian Ozsvald:
2. Parallel Programming: Techniques and Applications Using Networked Workstations and Parallel Computers" by Barry Wilkinson and Michael Allen:

YOUTUBE RESOURCES

1. **Corey Schafer** - Corey Schafer's channel offers comprehensive tutorials on Python programming, including topics related to parallel programming. His videos cover multiprocessing, multithreading, asynchronous programming with asyncio, and practical examples of parallelism in Python.
2. **sentdex** - Sentdex provides tutorials on various programming topics, including Python parallel programming. The channel covers multiprocessing, multithreading, and asynchronous programming concepts, with examples and demonstrations to help viewers understand the implementation of parallelism in Python.

TOTAL: 60 PERIODS

Course Code	EMPLOYABILITY ENHANCEMENT SKILLS - III	L	T	P	C
ES4301		0	0	2	1

COURSE OBJECTIVES:

The main objectives of this course are to:

- To give the exposure to solve the real time problems related to company test questions.
- 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:

CO1: To solve the real time programming problems related to company test questions.

CO2: To solve Quantitative Aptitude – Verbal Ability – Numerical Ability – Reasoning Ability Questions.

Semester IV

Course Code	COMPUTING THEORY AND COMPILER DESIGN	L	T	P	C
AD4401		3	0	0	3

COURSE OBJECTIVES:

1. Understand formal languages, regular expressions, and finite automata, and their application in lexical analysis.
2. Explore context-free grammars, parsing techniques, and ambiguity resolution for efficient language processing.
3. Analyze syntax-directed translation, intermediate code generation, and context-sensitive features in language semantics.
4. Evaluate storage allocation strategies, optimization techniques, and code generation algorithms for efficient program execution.
5. Implement machine-dependent code generation, object code forms, and register allocation to produce optimized executable code.

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.

45 PERIODS

COURSE OUTCOMES:

Upon completion of this course, the student should be able to:

CO1: Construct Finite Automata and write regular expression for any pattern.
CO2: Design context free grammar, Pushdown Automata and Turing Machine for computational functions.
CO3: Understand semantic rules and intermediate code generation.
CO4: Understand run-time environment and implement code generation.
CO5: Apply code optimization techniques.

TEXT BOOKS:

1. John E. Hopcroft, Rajeev M & J D Ullman: "Introduction to Automata Theory Languages &Computation", 3rd Edition, Pearson Education, 2007.
2. Aho, Ullman, Ravisethi: "Compilers Principles, Techniques, and Tools", 2nd Edition, Pearson Education, 2009.

REFERENCES:

1. Tremblay J P, Sorenson G P: "The Theory & Practice of Compiler writing", 1st Edition, BSP publication, 2010.
2. Appel W & Andrew G M: "Modern Compiler Implementation in C", 1st Edition, Cambridge University Press, 2003.
3. Louden: "Compiler Construction, Principles & Practice", 1st Edition, Thomson Press, 2006.
4. Sipser Michael: "Introduction to Theory of computation", 1st Edition, Thomson, 2009.

Course Code	BIG DATA ANALYTICS	L	T	P	C
AD4402		3	0	2	4

COURSE OBJECTIVES:

The main objectives of this course are to:

1. To Understand the Big Data Platform and its Use cases
2. To Provide the concept of the Hadoop framework and HDFS
3. To Understand Map-Reduce Jobs and Spark Framework
4. To Provide hands-on Hadoop Eco System
5. To provide Exposure to Data Analytics with R and Spark Shell

PREREQUISITES: Should have knowledge of one Programming Language (Java preferably), Practice of SQL (queries and sub queries), exposure to Linux Environment.

UNIT I INTRODUCTION TO BIG DATA ANALYTICS 9

Data Storage and Analysis: Types of Digital Data, Introduction to Big Data, Evolution, Characteristics of Big Data, Traditional Business Intelligence versus Big Data.

Big Data Analytics: Classification of Analytics, Big Data Analytics important, Typical Analytical Architecture – Requirement for new analytical architecture – Challenges in Big Data Analytics, Top Analytics Tools – Need of big data frameworks. Data Science - Data Scientist - Terminologies used in Big Data Environments.

UNIT II HADOOP FRAMEWORK AND HDFS (Hadoop Distributed File System) 9

Hadoop: History of Hadoop – Requirement of Hadoop Framework - Design principle of Hadoop – Comparison with other system - Hadoop Components – Hadoop 1 vs Hadoop 2 – Hadoop Daemon's – Apache Hadoop, Analysing, Data with Unix tools, Analysing Data with Hadoop, Hadoop Streaming.

HDFS: HDFS Concepts, HDFS Commands, Design of HDFS, Command Line Interface, Hadoop file system interfaces, Data flow, Data Ingest with Flume and Scoop and Hadoop archives, Hadoop I/O: Compression, Serialization, Avro and File-Based Data structures.

UNIT III MAP REDUCE AND SPARK FRAMEWORK 9

Map Reduce: Anatomy of a Map Reduce Job Run, Failures, Job Scheduling, Shuffle, and Sort, Task Execution, Map Reduce Types and Formats, Map Reduce Features. Map Reduce Programming: I/O formats, Map side join, Reduce Side Join, Secondary sorting, Pipelining MapReduce jobs.

Spark Framework: Introduction to GPU Computing, CUDA Programming Model, CUDA API, Simple Matrix, Multiplication in CUDA, CUDA Memory Model, Shared Memory Matrix Multiplication, Additional CUDA API Features.

UNIT IV HADOOP ECOSYSTEM 9

Introduction to Hadoop ecosystem technologies: Hadoop Echo System, Hadoop Streaming, Infosphere Big Insights, and Big Sheets. Scripting language: Introduction to PIG, Execution Modes of Pig, Comparison of Pig with Databases, Grunt, Pig Latin, User Defined Functions, Data Processing operators. Serialization: AVRO, Co-ordination: Zookeeper. Hive: Hive Shell, Hive

Services, Hive Metastore, Comparison with Traditional Databases, HiveQL, Tables, Querying Data and User-Defined Functions. HBase: HBasics, Concepts, Clients, Example, Hbase Versus RDBMS. Big SQL: Introduction. Streaming: Flink, Storm.

UNIT V DATA ANALYTICS 9

Data Analytics with R: Machine Learning: Introduction, Supervised Learning, Unsupervised Learning, Regression Model, Clustering, Collaborative Filtering, Associate Rule Making, Decision Tree, Big Data Analytics with BigR.
Data Analysis with Spark Shell: Writing Spark Application - Spark Programming in Scala, Python, R, Java - Application Execution.
Spark SQL and GraphX: SQL Context – Importing and Saving data – Data frames – using SQL – GraphX overview – Creating Graph – Graph Algorithms.
Spark Streaming: Overview – Errors and Recovery – Streaming Source – Streaming live data with spark.
Recent Trends in Big Data Analytics.

45 PERIODS

COURSE OUTCOMES:

Upon completion of the course, the students will be able to:
CO1: Identify Big Data and its Business Implications.
CO2: Understand the concept of Hadoop framework and HDFS
CO3: Understand the Map-Reduce Jobs and Spark Framework
CO4: Analyze Infosphere Big Insights Big Data Recommendations.
CO5: Apply Data Analytics with R and Spark Shell

PRACTICAL EXERCISES:

1. HDFS Commends Map Reduce Program to show the need of Combiner
2. Map Reduce I/O Formats-Text, key-value Map ReduceI/O Formats –Nline, Multiline
3. Sequence file Input/Output Formats Secondary sorting
4. Distributed Cache & Map Side Join, reduce side Join Building and Running a Spark Application Word count in Hadoop and Spark Manipulating RDD
5. Inverted Indexing in Spark Sequence alignment problem in Spark Implementation of Matrix algorithms in Spark Spark Sql programming, Building Spark Streaming application

15 PERIODS

TEXT BOOKS:

1. Tom White “Hadoop: The Definitive Guide” Third Edit on, O’reily Media, 2012.
2. Seema Acharya, Subhasini Chellappan, "Big Data Analytics" Wiley 2015.

REFERENCES:

1. Michael Berthold, David J. Hand, "Intelligent Data Analysis”, Springer, 2007.
2. Jay Liebowitz, “Big Data and Business Analytics” Auerbach Publications, CRC press (2013)
3. Tom Plunkett, Mark Hornick, “Using R to Unlock the Value of Big Data: Big Data Analytics with Oracle R Enterprise and Oracle R Connector for Hadoop”, McGraw-Hill/Osborne Media (2013), Oracle press.
4. Anand Rajaraman and Jefrey David Ulman, “Mining of Massive Datasets”, Cambridge University Press, 2012.

TOTAL: 60 PERIODS

Course Code	MACHINE LEARNING	L	T	P	C
AD4403		3	0	2	4

COURSE OBJECTIVES:

- To understand the basic concepts of machine learning.
- To understand and build supervised learning models.
- To understand and build unsupervised learning models.
- To learn the basics of deep learning using neural networks
- To learn the basics of optimization searching and reinforcement learning methods.

UNIT I INTRODUCTION **11**

Introduction: Introduction to Machine Learning: Introduction. Different types of learning, Hypothesis space and inductive bias, Evaluation. Training and test sets, cross validation, Concept of over fitting, under fitting, Bias and Variance. Linear Regression: Introduction, Linear regression, Simple and Multiple Linear regression, Polynomial regression, evaluating regression fit.

UNIT II SUPERVISED LEARNING **9**

Bayesian linear regression, gradient descent, Linear Classification Models: Discriminant function – Perceptron algorithm, Probabilistic discriminative model - Logistic regression, Probabilistic generative model – Naive Bayes, Maximum margin classifier – Support vector machine, Decision Tree, Random Forests, Instance Based Learning- KNN.

UNIT III NEURAL NETWORKS

Artificial Neural Networks: Introduction, Biological motivation, ANN representation, appropriate problem for ANN learning, Perceptron, multilayer networks and the back propagation algorithm, Popular CNN Architectures, RNNs, LSTM, BERT, GANS and Generative Models

UNIT IV UNSUPERVISED LEARNING AND OPTIMISATION **8**

Unsupervised learning: Expectation maximization - Gaussian mixture models -K-means / K medoid - hierarchal clustering-top-down, bottom-up –single linkage-multiple linkage. Dimensionality Reduction- Linear Discriminate Analysis, Principal Components Analysis, Factor Analysis, Independent Component Analysis. Optimization– Going Downhill, Least-Squares optimization, Conjugate Gradients - Exploitation and Exploration.

UNIT V APPLICATION PROBLEMS **9**

The case studies- churn analysis and prediction using Cox-proportional models, and churn prediction techniques. - Credit card fraud analysis with a focus on handling imbalanced data and neural networks. - Sentiment analysis and topic mining from the New York Times are addressed using similarity measures like cosine similarity, chi-square, and N-grams. - part-of-speech tagging, stemming, chunking - sales funnel analysis, A/B testing, and campaign effectiveness. - Web page layout effectiveness - recommendation systems with collaborative filtering - customer segmentation strategies and lifetime value- portfolio risk conformance and optimization, and Uber alternative routing with graph construction and route optimization.

Theory: 45 Periods

PRACTICAL EXERCISES:

1. Churn Analysis using Cox-Proportional Models: Implement Cox-Proportional Hazards models for customer churn prediction.

2. Credit Card Fraud Detection with Neural Networks: Build a neural network model to detect fraudulent transactions in an imbalanced dataset.
3. Sentiment Analysis of New York Times Articles: Perform sentiment analysis using cosine similarity, chi-square, and N-grams.
4. Part-of-Speech Tagging, Stemming, and Chunking: Apply POS tagging, stemming, and chunking techniques to a text dataset.
5. Sales Funnel Analysis and A/B Testing: Conduct sales funnel analysis and design A/B tests to evaluate marketing campaign effectiveness.
6. Web Page Layout Effectiveness Evaluation: Assess and improve web page layout effectiveness using analytical tools.
7. Collaborative Filtering Recommendation System: Create a recommendation system using user-based and item-based collaborative filtering.
8. Customer Segmentation and Lifetime Value Analysis: Use clustering algorithms for customer segmentation and analyze their lifetime value.
9. For a given set of training data examples stored in a .CSV file, implement and demonstrate the Candidate-Elimination algorithm to output a description of the set of all hypotheses consistent with the training examples.
10. Write a program to demonstrate the working of the decision tree based ID3 algorithm. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample.
11. Write a program to implement the naïve Bayesian classifier for a sample training data set stored as a .CSV file and compute the accuracy with a few test data sets.
12. Implement naïve Bayesian Classifier model to classify a set of documents and measure the accuracy, precision, and recall.
13. Write a program to construct a Bayesian network to diagnose CORONA infection using standard WHO Data Set.
14. Apply EM algorithm to cluster a set of data stored in a .CSV file. Use the same data set for clustering using the k-Means algorithm. Compare the results of these two algorithms.
15. Write a program to implement k-Nearest Neighbour algorithm to classify the iris data set. Print both correct and wrong predictions.
16. Implement the non-parametric Locally Weighted Regression algorithm in order to fit data points. Select an appropriate data set for your experiment and draw graphs.

Practical's: 30 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:

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

CO1: Explain the basic concepts of machine learning.
CO2: Construct supervised learning models.
CO3: Construct unsupervised learning algorithms.
CO4: Build neural network models
CO5: Build optimization searching and reinforcement models.

TEXTBOOKS:

1. Ethem Alpaydin, “Introduction to Machine Learning”, MIT Press, Fourth Edition, 2020.
2. Stephen Marsland, “Machine Learning: An Algorithmic Perspective, “Second Edition”, CRC Press, 2014.

REFERENCES:

1. Christopher M. Bishop, “Pattern Recognition and Machine Learning”, Springer, 2006.
2. Tom Mitchell, “Machine Learning”, McGraw Hill, 3rd Edition, 1997.
3. Mehryar Mohri, Afshin Rostamizadeh, Ameet Talwalkar, “Foundations of Machine Learning”, Second Edition, MIT Press, 2012, 2018.
4. Ian Goodfellow, Yoshua Bengio, Aaron Courville, “Deep Learning”, MIT Press, 2016
5. Sebastain Raschka, Vahid Mirjalili , “Python Machine Learning”, Packt publishing, 3rd Edition, 2019.

YOUTUBE RESOURCES:

Blue1Brown - This channel offers visually captivating explanations of various mathematical concepts, including those relevant to machine learning. Their series on neural networks and linear algebra is highly recommended for gaining a deeper understanding of the foundations behind machine learning algorithms.

sentdex - With a focus on Python programming and machine learning, this channel provides tutorials and walkthroughs on implementing machine learning algorithms using libraries like TensorFlow, scikit-learn, and PyTorch. It covers a wide range of topics, from basic concepts to advanced techniques.

Andrew Ng - Andrew Ng is a prominent figure in the machine learning community, and his lectures from courses like "Machine Learning" and "Deep Learning Specialization" are available on YouTube. These lectures provide a comprehensive introduction to machine learning and deep learning concepts.

StatQuest with Josh Starmer - This channel offers easy-to-understand explanations of statistical concepts, machine learning algorithms, and data analysis techniques. The videos are concise and visually engaging, making complex topics more accessible.

Brandon Rohrer - Brandon Rohrer's channel covers a variety of machine learning and data science topics, including tutorials on specific algorithms, discussions on best practices, and practical advice for working in the field.

TOTAL:75 PERIODS

Course Code	DATA WAREHOUSING AND DATA MINING	L	T	P	C
AD4404		3	0	2	4

COURSE OBJECTIVE

1. To understand data warehousing and mining principles.
2. To acquire familiarity with data warehouse architecture and implementation.
3. To explore data mining system architecture.
4. To comprehend various data preprocessing methods for mining tasks.
5. To develop proficiency in classification and prediction techniques.

UNIT I DATA WAREHOUSING AND BUSINESS ANALYSIS: 8

Data warehousing Components –Building a Data warehouse –Data Warehouse Architecture – DBMS Schemas for Decision Support – Data Extraction, Cleanup, and Transformation Tools – Metadata – reporting – Query tools and Applications – Online Analytical Processing (OLAP) – OLAP and Multidimensional Data Analysis.

UNIT II DATA CUBE TECHNOLOGY: 11

Data Cube Computation: Cube Materialization - Strategies for Data Cube Computation, Data Cube Computation Methods - Multiway Array Aggregation for Full Cube Computation - Star-Cubing: Computing Iceberg Cubes Using a Dynamic Star-Tree Structure - Precomputing Shell Fragments for Fast High-Dimensional OLAP, Processing Advanced Kinds of Queries by Exploring Cube Technology - Sampling Cubes: OLAP-Based - Ranking Cubes, Multidimensional Data Analysis in Cube Space - Prediction Cubes – Multi-feature Cubes - Exception-Based, Discovery-Driven Cube Space Exploration.

UNIT III INTRODUCTION TO DATA MINING: 8

Why Data Mining? - What Is Data Mining? - What Kinds of Data Can Be Mined? - What Kinds of Patterns Can Be Mined? - Which Technologies Are Used? - Which Kinds of Applications Are Targeted? - Major Issues in Data Mining - Data Objects and Attribute Types - Basic Statistical Descriptions of Data - Data Visualization - Measuring Data Similarity and Dissimilarity.

UNIT IV DATA PREPROCESSNG AND MINING TECHNIQUES: 10

Data Mining Functionalities – Data Preprocessing – Data Cleaning – Data Integration and Transformation – Data Reduction – Data Discretization and Concept Hierarchy Generation- Architecture of A Typical Data Mining Systems- Classification of Data Mining Systems.

Association Rule Mining: - Efficient and Scalable Frequent Item set Mining Methods – Mining Various Kinds of Association Rules – Association Mining to Correlation Analysis – Pattern Mining in Multilevel, Multidimensional Space - Constraint-Based Association Mining.

UNIT V MINING OBJECT, SPATIAL, MULTIMEDIA, TEXT AND WEB DATA: 8

Multidimensional Analysis and Descriptive Mining of Complex Data Objects – Spatial Data Mining – Multimedia Data Mining – Text Mining – Mining the World Wide Web.

45 PERIODS

LIST OF EXPERIMENTS 30 PERIODS

1. Design a data warehouse schema for a retail company to analyze sales data, customer demographics, and inventory management

2. Given a business scenario, use OLAP (Online Analytical Processing) to generate reports and perform trend analysis for strategic decision-making.
3. Using a dataset containing customer purchase history, apply association rule mining to discover patterns such as frequently co-purchased items.
4. Build and evaluate a predictive model for customer churn prediction using classification algorithms.
5. Utilizing a dataset of online user behaviour, apply sequence mining techniques to extract frequent sequences of website visits.
6. Given a dataset of patient medical records, preprocess the data and build a classifier to predict the likelihood of a patient developing a specific disease.
7. Describe the steps involved in feature selection for classification models. Identify and select relevant features from a high-dimensional dataset.
8. Using sentiment analysis on customer reviews as an example, evaluate the performance of a classification model and interpret the results.
9. Given a dataset of customer demographic information and purchasing behavior, use clustering algorithms to segment customers into distinct groups for targeted marketing campaigns.
10. Using a dataset of online user interactions with a website, use clustering to identify anomalous behavior or detect fraudulent activities.
11. Extract key features from images to perform object recognition using machine learning techniques?
12. Given a dataset of spatial coordinates representing locations of stores and customer residences, use spatial data mining to analyze geographical patterns and optimize store locations.
13. Using text mining, extract topics from a collection of news articles and visualize the relationships between different topics using network analysis techniques.

TOTAL PERIODS: 75

COURSE OUTCOMES:

At end of the course the students can
CO1: Enhanced business intelligence through optimized data storage, retrieval, and analysis.
CO2: Discovery of valuable patterns and insights from large datasets for informed decision-making.
CO3: Accurate categorization and prediction of data instances, facilitating personalized recommendations and risk assessment.
CO4: Identification of natural groupings within data for targeted marketing and anomaly detection.
CO5: Extraction of meaningful information from diverse data types, enabling comprehensive analysis and understanding.

TEXT BOOKS:

Jiawei Han, Micheline Kamber and Jian Pei"Data Mining Concepts and Techniques", Third Edition, Elsevier, 2011.

REFERENCES:

1. Alex Berson and Stephen J. Smith "Data Warehousing, Data Mining & OLAP", Tata McGraw – Hill Edition, Tenth Reprint 2007.
2. K.P. Soman, Shyam Diwakar and V. Ajay "Insight into Data mining Theory and Practice", Easter Economy Edition, Prentice Hall of India, 2006.
3. K. Gupta "Introduction to Data Mining with Case Studies", Easter Economy Edition, Prentice Hall of India, 2006.

Pang-Ning Tan, Michael Steinbach and Vipin Kumar "Introduction to Data Mining", Pearson Education, 2007.

Course Code	ARTIFICIAL INTELLIGENCE OF THINGS (AIoT)	L	T	P	C
AD4405		3	0	0	3

COURSE OBJECTIVES:

1. Understand the foundational concepts and significance of AIoT in various industries.
2. Learn machine learning and deep learning algorithms suitable for analyzing IoT data.
3. Integrate AI capabilities into IoT devices and understand edge computing architectures.
4. Design and implement end-to-end AIoT solutions for specific applications and use cases.
5. Identify security threats, privacy risks, and ethical concerns associated with AIoT deployments.

UNIT I INTRODUCTION TO ARTIFICIAL INTELLIGENCE OF THINGS (AIoT) 9

Understanding the convergence of artificial intelligence (AI) and Internet of Things (IoT) technologies. Exploring the evolution and significance of AIoT in various industries. Overview of AIoT architecture, components, and key concepts. Case studies highlighting real-world applications of AIoT.

UNIT II AI TECHNIQUES FOR IOT DATA ANALYSIS 9

Fundamentals of machine learning and deep learning algorithms for IoT data analysis. Preprocessing techniques for cleaning, transforming, and normalizing IoT data. Supervised learning methods (e.g., regression, classification) applied to IoT datasets for predictive analytics. Unsupervised learning techniques (e.g., clustering, anomaly detection) for extracting insights from IoT data. Introduction to reinforcement learning and its applications in optimizing IoT systems.

UNIT III IoT DEVICE INTEGRATION AND EDGE COMPUTING 9

Integrating AI capabilities into IoT devices and sensors. Understanding edge computing paradigms for efficient data processing and analysis at the network edge. Edge device architectures and frameworks for deploying machine learning models. Techniques for optimizing resource-constrained IoT devices for AI inference. Hands-on exercises on deploying AI models on edge devices using frameworks like TensorFlow Lite and ONNX Runtime.

UNIT IV AIoT SYSTEM DESIGN AND IMPLEMENTATION 9

Principles of designing AIoT systems for specific applications and use cases. Selection of hardware components, sensors, and communication protocols for AIoT deployments. Software development methodologies and frameworks for building AIoT applications. Design considerations for scalable, interoperable, and secure AIoT systems. Case studies and projects demonstrating the design and implementation of AIoT solutions.

UNIT V SECURITY, PRIVACY, AND ETHICAL CONSIDERATIONS IN AIoT 9

Identifying security threats and vulnerabilities in AIoT ecosystems. Privacy-preserving techniques for IoT data collection, storage, and processing. Compliance with regulations and standards (e.g., GDPR, HIPAA) governing AIoT deployments. Ethical considerations in AIoT, including bias, fairness, and transparency. Strategies for mitigating risks and ensuring trustworthiness in AIoT systems.

COURSE OUTCOMES:

At the end of the course, students have

CO1: Ability to explain the convergence of AI and IoT technologies and identify potential applications of AIoT.
CO2: Proficiency in applying supervised, unsupervised, and reinforcement learning techniques to extract insights and make predictions from IoT datasets.
CO3: Capability to deploy machine learning models on edge devices for real-time data processing and analysis, optimizing resource usage.
CO4: Ability to select appropriate hardware and software components, develop scalable AIoT architectures, and deploy functional systems.
CO5: Proficiency in implementing privacy-preserving techniques, ensuring compliance with regulations, and addressing ethical considerations to build trustworthy AIoT systems.

TEXT BOOKS:

1. "Artificial Intelligence of Things: Concepts and Applications" by Supriya K., Avinash S., and Rajesh M.
2. "Machine Learning for Internet of Things: Techniques, Tools, and Applications" by Samuel Son and Jong Hyuk Park
3. "Edge AI and Vision Intelligence: A Handbook for Next Generation IoT End Devices" by Victor Beagle
4. "Building Intelligent Systems: A Guide to Machine Learning Engineering" by Geoff Hulten
5. "AI Ethics: An Introduction" by Markus Dubber

Course Code	COMPUTER NETWORKS	L	T	P	C
AD4406		3	0	2	4

COURSE OBJECTIVES:

The main objectives of this course are to:

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

COURSE DESCRIPTION:

Explore the intricacies of advanced data link and network layer technologies, delve into optimization strategies for transport and application layers, and master cutting-edge concepts in network security. The hands-on experience in this course prepares students for the challenges of modern networking, including emerging technologies like blockchain and zero trust networking.

PREREQUISITES:

Basics of networks, OSI model, and common networking protocols required.
Proficiency in one language (Python), basic socket programming understanding preferred.

COURSE SYLLABUS:

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 EXERCISES **30 PERIODS**

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 the Simulation tool.
9. Simulation of Distance Vector/Link State Routing algorithm.
10. Simulation of an error correction code (like CRC**)**

TOTAL: 75 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: Explain the significance of computer networks in modern computing.

CO2: Analyze the efficiency and performance of different data link layer protocols.

CO3: Implement advanced concepts in the network layer by configuring and troubleshooting IP routing, routing protocols (e.g., OSPF, BGP), and multicast routing.

CO4: Analyze the performance of transport and application layer protocols, utilizing advanced congestion control algorithms and optimization techniques.

CO5: Evaluate the implementation and effectiveness of network security technologies.

TEXT BOOKS:

1. "Computer Networking: A Top-Down Approach" by James F. Kurose and Keith W. Ross, Latest edition published in 2017
2. "Computer Networking: Principles, Protocols and Practice" by Olivier Bonaventure, First edition published in 2012 (open-access)
3. "Computer Networks" by Andrew S. Tanenbaum and David J. Wetherall, Latest edition published in 2020.

4. "Data Communications and Networking" by Behrouz A. Forouzan, Latest edition published in 2020

REFERENCES:

1. "Computer Networking Problems and Solutions: An innovative approach to building resilient, modern networks" by Russ White and Ethan Banks, Published in 2020
2. "Network Security Essentials" by William Stallings, Latest edition published in 2021.
3. "SDN: Software Defined Networks" by Thomas D. Nadeau and Ken Gray, Published in 2013.
4. "TCP/IP Illustrated, Volume 1: The Protocols" by W. Richard Stevens, Latest edition published in 2011.
5. "High-Performance Browser Networking" by Ilya Grigorik, Published in 2013.

E-BOOK:

1. An Introduction to Computer Networks Peter L Doral First – 2020
   http://intronetowrks.cs.luc.edu/current/ComputerNetworks

YOUTUBE REFERENCES:

1. PowerCert Animated Videos**:** PowerCert provides animated videos explaining various networking concepts, protocols, and technologies.
2. David Bombal**:** David Bombal's channel offers tutorials on networking, Cisco certifications, and hands-on labs.
3. Eli the Computer Guy**:** Eli the Computer Guy covers a wide range of IT and networking topics, including practical advice and troubleshooting.
4. Hak5**:** Hak5 explores technology and security, including networking topics and tools.

WEBSITES:

1. **Cisco Networking Academy:** Cisco Networking Academy provides online courses, interactive tools, and resources for learning about networking, including hands-on labs and simulations.
2. **Wireshark:** Wireshark is a widely used network protocol analyzer. The website offers documentation, tutorials, and resources for learning how to use Wireshark for network analysis.
3. **IETF - Internet Engineering Task Force:** IETF is the organization that develops and promotes Internet standards. The site provides access to RFCs (Request for Comments) and other documents related to networking protocols.
4. **Internet Society (ISOC):** Internet Society is a global organization dedicated to ensuring the open development, evolution, and use of the Internet. Their resources cover various aspects of networking.
5. **Network World:** Network World is a news and information website covering the field of networking. It provides articles, analysis, and insights into the latest trends and technologies.

Course Code	EMPLOYABILITY ENHANCEMENT SKILLS - IV	L	T	P	C
ES4401		0	0	2	1

COURSE OBJECTIVES:

- The main objectives of this course are to:
- To give the exposure to solve the real time problems related to company test questions.
- 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:

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

Semester V

Course Code	NATURAL LANGUAGE PROCESSING	L	T	P	C
AD4501		3	0	2	4

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

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.

TEXT BOOKS:

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	DEEP LEARNING	L	T	P	C
AD4502		3	0	2	4

COURSE OBJECTIVES:

The main objectives of this course are to:

- To understand the basic ideas and principles of neural networks and concepts of deep learning.
- To study Convolutional Neural Networks with image processing facilities like TensorFlow and Keras.
- To study Recurrent Neural Networks with speech processing models
- To study Deep Reinforcement Learning and the use of real-time applications.
- To understand and implement deep learning architectures.

COURSE DESCRIPTION:

"Deep Learning" delves into advanced neural network architectures and algorithms for machine learning tasks. Topics include deep neural networks, convolutional neural networks (CNNs), recurrent neural networks (RNNs), and their applications in natural language processing, computer vision, and more.

PREREQUISITES

Proficiency in programming languages such as Python.
Understanding of basic machine learning concepts and algorithms.
Familiarity with linear algebra, calculus, and probability theory.
Knowledge of data preprocessing and feature engineering techniques.
Experience with libraries such as TensorFlow or PyTorch is beneficial.

UNIT I FUNDAMENTALS OF DEEP LEARNING 9

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 9

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 9

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 9

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 9

Autonomous Vehicles Introduction – Imitation driving policy – Driving policy with ChaufferNet – DL in Cloud

45 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:

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

CO1: Understanding the basics concepts of deep learning.
CO2: Emphasizing knowledge of Convolutional Neural Networks and applying CNN to its variants for suitable application.
CO3: Understanding of Recurrent Neural Networks to apply autoencoders and generative models for suitable applications.
CO4: Understanding deep reinforcement learning and to
CO5: Analyze the key computations underlying deep learning and use them to build and train deep neural networks for various tasks.

TEXT BOOKS:

1. Eugene Charniak, “Introduction to Deep Learning”, MIT Press, 2018.
2. Ivan Vasiliev, 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. Ethem Alpaydin, "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	REINFORCEMENT LEARNING	L	T	P	C
AD4503		3	1	0	4

OBJECTIVE

- Understand the basic concepts of reinforcement learning, including agents, environments, and rewards.
- Learn dynamic programming methods for solving reinforcement learning problems
- Master Monte Carlo methods for estimating value functions and improving policies based on experience.
- Understand temporal difference learning as a combination of Monte Carlo and dynamic programming methods.
- Explore deep reinforcement learning techniques and their applications.

UNIT I INTRODUCTION TO REINFORCEMENT LEARNING **15**

Theory:

Overview of reinforcement learning (RL) concepts, including agents, environments, and rewards- Markov Decision Processes (MDPs) as a formal framework for modeling RL problems- Exploration vs. exploitation trade-off and the concept of policy in RL.

Practical:

1. Implementing basic RL algorithms such as multi-armed bandits and Q-learning in Python.
2. Applying RL techniques to simple grid world environments for learning optimal policies.

UNIT II DYNAMIC PROGRAMMING IN REINFORCEMENT LEARNING **15**

Theory:

Introduction to dynamic programming (DP) methods for solving RL problems - Policy evaluation, policy improvement, and policy iteration algorithms - Value iteration and policy iteration algorithms for solving finite MDPs.

Practical:

1. Implementing dynamic programming algorithms for solving grid world and maze navigation problems.
2. Applying DP techniques to approximate optimal policies in larger MDPs.

UNIT III MONTE CARLO METHODS **15**

Theory:

Introduction to Monte Carlo methods for estimating value functions and improving policies - Monte Carlo prediction and control algorithms for learning from experience - Exploring state and action spaces using Monte Carlo tree search (MCTS).

Practical:

1. Implementing Monte Carlo methods for solving Blackjack and other episodic tasks.
2. Applying Monte Carlo tree search to games like Tic-Tac-Toe and Go.

UNIT IV TEMPORAL DIFFERENCE LEARNING **15**

Theory:

Temporal difference (TD) learning as a combination of Monte Carlo and dynamic programming methods - TD prediction (TD(0)) and TD control (SARSA and Q-learning) algorithms - Eligibility traces and n-step TD methods for bootstrapping in RL.

Practical:

Implementing TD learning algorithms for solving Cliff Walking and Mountain Car problems.
Applying TD methods to learn optimal policies in simulated robotic control tasks.

UNIT V DEEP REINFORCEMENT LEARNING **15**

Theory:

Introduction to deep reinforcement learning (DRL) and its applications - Deep Q-Networks (DQN) for approximating Q-values using neural networks - Policy gradient methods such as REINFORCE and Actor-Critic for learning parameterized policies.

Practical:

Implementing DQN algorithm to play Atari games using OpenAI Gym.

Training policy gradient models to solve continuous control tasks like robotic manipulation.

TOTAL PERIODS: 75

COURSE OUTCOMES:

At the end of the course, Students have

CO1: Ability to describe the components of a reinforcement learning problem and its applications in various domains.

CO2: Proficiency in implementing policy evaluation, policy improvement, and value iteration algorithms to find optimal policies.

CO3: Capability to implement Monte Carlo prediction and control algorithms for learning optimal policies in episodic tasks.

CO4: Proficient implementation of TD prediction and control algorithms such as SARSA and Q-learning for learning from incomplete sequences of experience.

CO5: Ability to implement deep Q-networks (DQN) and policy gradient methods for solving complex reinforcement learning tasks with high-dimensional state spaces.

TEXT BOOK:

1. "Reinforcement Learning: An Introduction" by Richard S. Sutton and Andrew G. Barto
"Deep Reinforcement Learning Hands-On" by Maxim Lapan

Semester VI

Course Code	CLOUD AND DISTRIBUTED COMPUTING	L	T	P	C
AD4601		3	0	2	4

COURSE OBJECTIVES:

The main objectives of this course are to:

- The objective of this course is to enable the students to
- Explain about distributed system and cloud models.
- Apply distributed computational model and understand the need for cloud computing.
- Learn about cloud recovery and storage.

UNIT I INDRODUCTION TO DISTRIBUTED SYSTEM 6

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 DISTRIBUTED ALGORITHMS 6

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 6

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 6

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 6

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

30 PERIODS

COURSE OUTCOMES:

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

CO1: Understand how distributed system concepts apply inside cloud.

CO2: Apply the distributed algorithms for effective scheduling.

CO3: Analyze the need for virtualization and apply it in a cloud environment.

CO4: Understand the architecture, infrastructure, and delivery models of cloud computing.
CO5: Explore cloud tools and build applications.

PRACTICAL EXERCISE:

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

30 PERIODS

TEXT BOOKS:

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:

1. 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.

TOTAL: 60 PERIODS

Course Code	GENERATIVE AI	L	T	P	C
AD4602		2	0	2	3

COURSE OBJECTIVES:

The main objectives of this course are to:

- Understand the principles and theory behind generative AI.
- Gain practical experience in developing generative AI models.
- Analyze and evaluate the ethical and societal implications of generative AI.
- Apply generative AI techniques to real-world problems and domains.
- Keep up-to-date with the latest developments and trends in the field of generative AI.

Course Description

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:

1. Ian Goodfellow, Yoshua Bengio, Aaron Courville, "Deep Learning", The MIT Press, 2016.
2. Alberto Chierici, "The Ethics of AI", New Degree Press, 2021.
4. Jacob Emerson, "Ripples of Generative AI", IngramSpark, 2023.
5. Francois Chollet, "Deep Learning with Python", Second Edition, Manning, 2021.
6. 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.
3. Two Minute Papers - This channel offers concise summaries of research papers and breakthroughs in computer graphics, machine learning, and AI, including generative AI.
4. DeepLizard - This channel provides tutorials on machine learning and deep learning topics, which can be useful for understanding the technical aspects of generative AI.

5. Andrew Ng's Deeplearning.ai - The deeplearning.ai courses on Coursera, created by Andrew Ng, are also available on YouTube. These videos cover deep learning and neural networks, which are foundational to generative AI.
6. Stanford University's Machine Learning - You can find recordings of Stanford University's machine learning course on YouTube, which includes lectures on relevant topics. (Stanford Online)
7. PyTorch - The official PyTorch YouTube channel provides tutorials and resources for learning PyTorch, a popular framework for deep learning and generative AI.
8. TensorFlow - The official TensorFlow YouTube channel offers tutorials and resources for learning TensorFlow, another widely used deep learning framework.

45 PERIODS

TOTAL:75 PERIODS

Course Code	COMPUTER VISION	L	T	P	C
AD4603		2	0	2	3

COURSE OBJECTIVES:

The main objectives of this course are to:

1. To understand the fundamental concepts related to Image formation and processing.
2. To learn feature detection, matching, and detection
3. To become familiar with feature-based alignment and motion estimation
4. To develop skills in 3D reconstruction
5. To understand image-based rendering and recognition

UNIT I INTRODUCTION TO IMAGE FORMATION AND PROCESSING 6

Computer Vision - Geometric primitives and transformations - Photometric image formation - The digital camera - Point operators - Linear filtering - More neighborhood operators - Fourier transforms - Pyramids and wavelets - Geometric transformations - Global optimization.

UNIT II FEATURE DETECTION, MATCHING AND SEGMENTATION 6

Points and patches - Edges - Lines - Segmentation - Active contours - Split and merge - Mean shift and mode finding - Normalized cuts - Graph cuts and energy-based methods.

UNIT III FEATURE-BASED ALIGNMENT & MOTION ESTIMATION 6

2D and 3D feature-based alignment - Pose estimation - Geometric intrinsic calibration - Triangulation - Two-frame structure from motion - Factorization - Bundle adjustment - Constrained structure and motion - Translational alignment - Parametric motion - Spline-based motion - Optical flow - Layered motion.

UNIT IV 3D RECONSTRUCTION 6

Shape from X - Active range finding - Surface representations - Point-based representations- Volumetric representations - Model-based reconstruction - Recovering texture maps and albedo.

UNIT V IMAGE-BASED RENDERING AND RECOGNITION 6

View interpolation Layered depth images - Light fields and Lumigraphs - Environment mattes - Video-based rendering-Object detection - Face recognition - Instance recognition - Category recognition - Context and scene understanding- Recognition databases and test sets.

30 PERIODS

COURSE OUTCOMES:

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

CO1: To understand basic knowledge, theories, and methods in image processing and computer vision.
CO2: To implement basic and some advanced image processing techniques in OpenCV.
CO3: To apply 2D a feature-based image alignment, segmentation, and motion estimations.
CO4: To apply 3D image reconstruction techniques
CO5: To design and develop innovative image processing and computer vision applications.

PRACTICAL EXERCISES: TOTAL: 30 PERIODS

Software needed:

1. OpenCV computer vision Library for OpenCV in Python / PyCharm or C++ / Visual Studio or equivalent
2. OpenCV Installation and working with Python
3. Basic Image Processing - loading images, Cropping, Resizing, Thresholding, Contour analysis, Bolb detection
4. Image Annotation – Drawing lines, text circle, rectangle, ellipse on images
5. Image Enhancement - Understanding Color spaces, color space conversion, Histogram equalization, Convolution, Image smoothing, Gradients, Edge Detection
6. Image Features and Image Alignment – Image transforms – Fourier, Hough, Extract ORB Image features, Feature matching, cloning, Feature matching based image alignment
7. Image segmentation using Graphcut / Grabcut
8. Camera Calibration with a circular grid
9. Pose Estimation
10. 3D Reconstruction – Creating Depth map from stereo images
11. Object Detection and Tracking using Kalman Filter, Camshift

Resoursces:

1. docs.opencv.org
2. https://opencv.org/opencv-free-course/

TOTAL: 60 PERIODS

TEXT BOOKS:

1. Richard Szeliski, “Computer Vision: Algorithms and Applications”, Springer- Texts in Computer Science, Second Edition, 2022.
2. Computer Vision: A Modern Approach, D. A. Forsyth, J. Ponce, Pearson Education, Second Edition, 2015.

REFERENCES:

1. Richard Hartley and Andrew Zisserman, Multiple View Geometry in Computer Vision, Second Edition, Cambridge University Press, March 2004.
2. Christopher M. Bishop; Pattern Recognition and Machine Learning, Springer, 2006
3. E. R. Davies, Computer and Machine Vision, Fourth Edition, Academic Press, 2012.

Semester VII

Course Code	PROFESSIONAL ETHICS AND UNIVERSAL HUMAN VALUES	L	T	P	C
HS4701		3	0	0	3

COURSE OBJECTIVE:

To enable the students to create an awareness of Engineering Ethics and Human Values, to still 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:

Upon completion of the course, the student should be able to apply ethics in society, discuss the ethical issues related to engineering, and realize societal responsibilities and rights.

TEXT BOOKS:

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 Cloud Computing	Vertical V IoT Systems	Vertical VI High-End Computing
AD4V11 Web Development 5.0	**AD4V21** Software Engineering	**AD4V31** Cryptography & Information Security	**AD4V41** Cloud Technologies	**AD4V51** Foundations of Embedded IoT Systems	**AD4V61** Parallel Processing
AD4V12 App Development	**AD4V22** Software Testing and Automation	**AD4V32** Security and Privacy in Cloud	**AD4V42** Virtualization	**AD4V52** IoT Networks	**AD4V62** Hi-Performance Computing
AD4V13 UI and UX Design	**AD4V23** Agile Methodologies	**AD4V33** Web Application Security	**AD4V43** Cloud Architectures	**AD4V53** Secure Hardware and Embedded Devices	**AD4V63** Pervasive Computing
AD4V14 Cloud Services Management	**AD4V24** Software User Interface Design & Analysis	**AD4V34** Social Network Security	**AD4V44** Cloud Platform Programming	**AD4V54** IoT Processors	**AD4V64** Pico Computing
AD4V15 DevOps	**AD4V25** Software Architecture and Design Patterns	**AD4V35** Digital Forensics and Malware Analysis	**AD4V45** Cloud Computing	**AD4V55** Mobile Applications Development	**AD4V65** Nano Computing
AD4V16 Advanced Web Frameworks and Containerization	**AD4V26** Software Quality Management	**AD4V36** Ethical Hacking	**AD4V46** Stream Processing	**AD4V56** Industrial IoT & Healthcare Systems	**AD4V66** Edge and Fog Computing
AD4V17 Middle Tier Technologies	**AD4V27** Software Project Management	**AD4V37** Cryptocurrency & Blockchain Technologies	**AD4V47** Fog and Edge Computing	**AD4V57** Smart Cities	**AD4V67** AI and Cloud Computing
AD4V18 Web Application Security	**AD4V28** Human-Computer Interaction	**AD4V38** Security Auditing & Counter Hacking Techniques	**AD4V48** Blockchain Concepts and Technologies	**AD4V58** Advanced Intelligent Systems	**AD4V68** 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.

APPENDIX A: PROFESSIONAL ELECTIVES

VERTICAL I: FULL STACK DEVELOPMENT

Course Code	WEB DEVELOPMENT 5.0	L	T	P	C
AD4V11		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 1: 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 2: 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 3: 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 4: 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 5: 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

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.

TEXT BOOKS:

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
AD4V12		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

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 current location of the user.
7. Write programs using Java to create Android application having Databases
8. For a simple library application.
9. 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.

TEXT BOOKS:

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, Full Stack 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
AD4V13		2	0	2	3

COURSE OBJECTIVES:

To provide a sound knowledge in UI & UX
To understand the need for UI and UX
To understand the various Research Methods used in Design
To explore the various Tools used in UI & UX
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

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 Wireflow 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. 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

TEXT BOOKS:

1. Joel Marsh, "UX for Beginners", O'Reilly , 2022
2. Jon Yablonski, "Laws of UX using Psychology to Design Better Product & Services" O'Reilly2021

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

Course Code	CLOUD SERVICES MANAGEMENT	L	T	P	C
AD4V14		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

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.

TEXT BOOKS:

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	DevOps	L	T	P	C
AD4V15		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

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

TEXT BOOKS:

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.

Course Code	ADVANCED WEB FRAMEWORKS AND	L	T	P	C
AD4V16	CONTAINERIZATION	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

UNIT IV REACT HOOKS **6**

React Hooks – useState – useEffect – useCallback – useMemo – useContext – useReducer – Introduction to React Native

UNIT V CONTAINERIZATION **6**

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**

Project – Create an angular app with n components and add routing
Project – Add functionalities, validation and database with above components
Project – Create Login System using React
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

TEXT BOOKS:

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.

Course Code	MIDDLE TIER TECHNOLOGIES	L	T	P	C
AD4V17		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

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

TEXT BOOKS:

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
AD4V18		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 II SECURE 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 V HACKING 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

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

TEXT BOOKS:

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.

VERTICAL II: SOFTWARE TECHNOLOGIES

Course Code	SOFTWARE ENGINEERING	L	T	P	C
AD4V21		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 – Petrinets – 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

NUMBER OF THEORY PERIODS: **30**

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
6. UML Sequence and Collaboration Diagrams

7. Draw relevant State Chart and Activity Diagrams for the same system.
8. Implement the system as per the detailed design
9. Test the software system for all the scenarios identified as per the use case diagram
10. Improve the reusability and maintainability of the software system by applying appropriate design patterns.
11. Implement the modified system and test it for various scenarios.

SUGGESTED DOMAINS FOR MINI-PROJECTS:

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

TEXT BOOKS:

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
3. 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
AD4V22		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 IITEST 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 VTEST 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

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:
10. Build a data-driven framework using Selenium and TestNG
11. Build Page object Model using Selenium and TestNG
12. 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

TEXT BOOKS:

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
AD4V23		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

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

1. Understand the background and driving forces for taking an Agile Approach to Software development.
2. Build out a backlog and user stories.
3. To study automated build tool.
4. To study version control tool.
5. To study Continuous Integration tool.
6. Apply Design principle and Refactoring to achieve agility.
7. 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.

TEXT BOOKS:

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 &	L	T	P	C
AD4V24	ANALYSIS	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 I 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 II 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 III 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 IV 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 V 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

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.

TEXT BOOKS:

1. "Don't Make Me Think" by Steve Krug
2. "The Design of Everyday Things" by Don Norman
3. "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	L	T	P	C
AD4V25	PATTERNS	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

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.

TEXT BOOKS:

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.”Microservices Patterns: With Examples in Java" by Chris Richardson
2."Building Micro services" by Sam Newman
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
AD4V26		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 I 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 II 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 III 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 IV 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 V 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

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.

TEXT BOOKS:

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
AD4V27		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

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

TEXT BOOK:

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.
2. Walker Royce: —Software Project Management- Addison-Wesley, 1998. 3. Gopalaswamy Ramesh, —Managing Global Software Projects – McGraw Hill Education (India), Fourteenth Reprint 2013.

Course Code	HUMAN-COMPUTER INTERACTION	L	T	P	C
AD4V28		2	0	2	3

COURSE OBJECTIVES

To learn the foundations of Human Computer Interaction.
To become familiar with the design technologies for individuals and persons with disabilities.
To be aware of mobile HCI.
To learn the guidelines for user interface.
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

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.

TEXT BOOKS:

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
AD4V31		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, STEGANOGRAPHY.

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

TEXT BOOKS:

1. William Stallings (2006), Cryptography and Network Security: Principles and Practice, 4th edition, Pearson Education, India.
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. 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
AD4V32		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 PERIODS: 30

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.

TEXT BOOKS:

1. Raj Kumar Buyya , James Broberg, andrzejGoscinski, "Cloud Computing:", Wiley 2013
2. Dave shackleford, "Virtualization Security", SYBEX a wiley Brand 2013.
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
AD4V33		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. cross-site scripting (XSS)
   b. SQL injection
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

TEXT BOOKS:

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
AD4V34		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

TEXT BOOKS:

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
AD4V35		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.

TEXT BOOKS:

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

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

Online Reference:

https://www.coursera.org/specializations/computerforensics
https://www.youtube.com/watch?v=u2zgEFm5RHQ
Practical Malware analysis - Youtube

Course Code	ETHICAL HACKING	L	T	P	C
AD4V36		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

TEXT BOOKS:

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. 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
AD4V37		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 cryptocurrency domain.

CO3 It provides a conceptual understanding of the function of Blockchain as a method of securing distributed ledgers.

CO4 Apply hyper ledger Fabric and Ethereum platform to implement the Block chain Application.

CO5 To design and develop the Blockchain Application.

TEXT BOOKS:

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
AD4V38	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

TEXT BOOKS:

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 Mandia (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)

VERTICAL IV: CLOUD COMPUTING

Course Code	CLOUD TECHNOLOGIES	L	T	P	C
AD4V41		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 opensource 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.

TEXT BOOKS:

1. Kai Hwang, Geoffrey C Fox, Jack G Dongarra, "Design and Analysis of Algorithms, 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
AD4V42		2	0	2	3

COURSE OBJECTIVES:

1. To Learn the basics and types of Virtualizations
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.

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
e) Desktop Virtualization using VNC
f) Desktop Virtualization using Chrome Remote Desktop

2. Create type 2 virtualization on ESXI 6.
3. 5 server 5. Create a VLAN in CISCO packet tracer
4. Install KVM in Linux
   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 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
AD4V43		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

TEXT BOOKS:

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
AD4V44		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-native 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. **Setting Up a Cloud Account:**
   a. Create an account on a major cloud platform such as AWS, Azure, Google Cloud, or others.
   b. Explore the platform's dashboard and understand the basic services offered.
2. **Virtual Machines and Containers:**
   a. Launch a virtual machine (VM) instance on the cloud platform.
   b. Deploy a containerized application using a container orchestration tool like Docker or Kubernetes.
3. **Serverless Computing:**
   a. Create a serverless function using platforms like AWS Lambda, Azure Functions, or Google Cloud Functions.
   b. Trigger the function in response to an event (e.g., HTTP request, file upload, database change).
4. **Cloud Storage:**
   a. Use cloud storage services (e.g., Amazon S3, Azure Blob Storage) to store and retrieve files.
   b. Implement versioning, access control, and lifecycle policies for stored objects.

5.	**Database Services:**
a.	Set up a managed database instance (e.g., AWS RDS, Azure SQL Database).
b.	Connect your application to the database and perform CRUD operations.
6.	**Networking and Security:**
a.	Configure a virtual network with subnets, security groups, and network ACLs.
b.	Implement SSL/TLS for securing communication between components.
7.	**Monitoring and Logging:**
a.	Set up monitoring for your cloud resources using tools like AWS CloudWatch, Azure Monitor, or Google Cloud Monitoring.
b.	Configure logging to track and analyze application and infrastructure logs.
8.	**Continuous Integration/Continuous Deployment (CI/CD):**
a.	Implement a CI/CD pipeline using services like AWS CodePipeline, Azure DevOps, or Google Cloud Build.
b.	Automate the build, test, and deployment processes for your application.
9.	**Scaling and Load Balancing:**
a.	Configure auto-scaling policies for your application based on metrics like CPU utilization.
b.	Set up a load balancer to distribute incoming traffic across multiple instances.
10.	**Identity and Access Management (IAM):**
a.	Define IAM roles and policies to control access to your cloud resources.
b.	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.

TEXT BOOKS:

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)
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 COMPUTING	L	T	P	C
AD4V45		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 **7**

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 **5**

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.

30 PERIODS

PRACTICAL EXERCISES: **30 PERIODS**

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:

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.

TOTAL: 60 PERIODS

TEXT BOOKS

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	STREAM PROCESSING	L	T	P	C
AD4V46		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

UNIT IV EVENT PROCESSING WITH APACHE KAFKA **6**

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 an 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.

TEXT BOOKS:

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

Course Code	FOG AND EDGE COMPUTING	L	T	P	C
AD4V47		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 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
AD4V48		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 conceptsto 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 analyze 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 Architecture Design:** 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. **DomainName 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.

TEXT BOOKS:

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 and Cryptocurrency 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.

VERTICAL V: IOT SYSTEMS

Course Code	FOUNDATIONS OF EMBEDDED IOT SYSTEMS	L	T	P	C
AD4V51		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 Pi embedded 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 Systems.

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 BOOKS:

1. Muhammad Ali Mazidi Shujen Chen, Sepehr Naimi Sarmad Naimi "Embedded Programming Using C Language", 1st Edition, Freescale ARM Cortex-M.
2. Steve Ferbur, "ARM System on Chip". Rajkamal, "Embedded System: Architecture, Programming and Design", TMH3.
3. Dr. OvidiuVermesan, Dr. Peter Friess, "Internet of Things: Converging Technologies for Smart Environments and Integrated Ecosystems", River Publisher

Course Code	IOT NETWORKS	L	T	P	C
AD4V52		2	0	2	3

COURSE OBJECTIVES:

1. To give an understanding of 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 applications 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 to Chapters 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 to Chapters 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 to Chapters 1,2, 3 of Text 2.

UNIT V COMMUNICATION PROTOCOLS **6**

Physical Layer and Transceiver Design Considerations, MAC Protocols for Wireless 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 to Chapters 4, 5, 7, and 11 of Text 2. Lt, L2, L3

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

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
AD4V53		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

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 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
AD4V54		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 Assembler 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

a) Write a program to turn on green LED (Port B.6) and Blue LED (Port B.7) on STM32L-Discovery by configuring GPIO.
b) Transmit a string "Programming with ARM Cortex" to PC by configuring the registers of USART2. Use polling method.

4. ARM Cortex M3/M4 Programming with CMSIS
   a) Write a program to toggle the LEDs at the rate of 1 sec using standard peripheral library. Use Timer3 for Delay.
   b) 2. 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 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. 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
AD4V55		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 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
AD4V51		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 REQUIREMENTS **6**

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**

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.

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

TEXT 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
AD4V57		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.

TEXT BOOKS:

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
AD4V58		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 optimizationfor 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 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)

VERTICLE VI: HIGH END COMPUTING

Course Code	PARALLEL PROCESSING	L	T	P	C
AD4V61		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

TOTAL PERIODS :45

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
AD4V62		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 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
AD4V63		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
AD4V64		3	0	0	3

COURSE OBJECTIVES:

1. Understanding PICO Computing Principles
2. Exploring PICO Architecture
3. Mastering PICO Programming Skills
4. Examining PICO Operating Systems
5. Investigating Advanced Topics in PICO Computing

UNIT 1: INTRODUCTION TO PICO COMPUTING: **9**

Overview of PICO Computing – Definition and characteristics, Distinction from traditional computing. Historical perspective and evolution – Milestones in PICO Computing development, Key contributors and innovations. Importance and applications in modern computing – Use cases and real-world examples, Advantages and limitations of PICO Computing. Characteristics of PICO Computing systems – Scalability and flexibility, Energy efficiency and power consumption, Reliability and fault tolerance.

UNIT II PICO ARCHITECTURE **9**

Basic principles of PICO architecture – Instruction set architecture (ISA), Data and control flow in PICO systems. Components and building blocks – Processors, memory, and interconnects, Custom hardware accelerators. PICO processors and memory – Types of PICO processors (e.g., PICO CPUs, GPUs), Memory hierarchy and organization. Parallelism and concurrency in PICO Computing – Parallel processing models, Synchronization and communication mechanisms, Task parallelism vs. data parallelism

UNIT III PROGRAMMING FOR PICO COMPUTING **9**

PICO programming languages – High-level languages for PICO systems, Low-level languages and assembly for PICO architectures. PICO-specific programming paradigms – Data parallel programming, Task-based programming models. Code optimization techniques for PICO systems – Compiler optimizations, Manual optimization strategies. Case studies and examples – Practical applications of PICO programming, Challenges and solutions in PICO software development.

UNIT IV PICO OPERATING SYSTEMS **9**

Role of operating systems in PICO Computing – Resource management and abstraction, PICO OS vs. traditional OS. PICO OS design principles – Minimalist kernel design, Security considerations in PICO OS. Kernel architecture for PICO systems – System calls and kernel services, Device drivers for PICO peripherals. Resource management and scheduling in PICO OS - Memory allocation and deallocation, Scheduling algorithms for PICO processors.

UNIT V ADVANCED TOPICS IN PICO COMPUTING **9**

Emerging trends in PICO Computing - Edge computing and PICO devices, Quantum-inspired PICO architectures. Quantum aspects of PICO systems – Quantum computing principles in PICO, Quantum algorithms and their implications. PICO Computing in specific domains – Healthcare applications, financial modeling and PICO, AI and machine learning on PICO systems. Ethical considerations and challenges in PICO Computing – Privacy concerns in PICO applications, social implications of PICO technology.

NUMBER OF THEORY PERIODS: 45

COURSE OUTCOMES:

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

CO1 Demonstration of Comprehensive PICO Computing Knowledge
CO2 Application of PICO Architecture Concepts
CO3 Proficient PICO Programming Skills Development
CO4 Design and Analysis of PICO Operating Systems
CO5 Critical Evaluation of Advanced PICO Computing Concepts

TEXTBOOKS:

1. “Parallel Computer Architecture: A Hardware/Software Approach” by David Culler, Jaswinder Pal Singh, and Anoop Gupta
2. "Quantum Computing: A Gentle Introduction" by Eleanor G. Rieffel and Wolfgang H. Polak - Although focused on quantum computing.
3. "Computer Organization and Design: The Hardware/Software Interface" by David A. Patterson and John L. Hennessy.
4. “Programming Massively Parallel Processors: A Hands-on Approach" by David B. Kirk and Wen-mei W. Hwu -
5. “Operating System Concepts" by Abraham Silberschatz, Greg Gagne, and Peter B.
6. “Edge Computing: Models, Technologies, and Applications" by Danda B. Rawat, Joel J.P.C. Rodrigues, and Ivan Stojmenovic.

Course Code	NANO COMPUTING	L	T	P	C
AD4V65		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
AD4V66		3	0	0	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 **9**

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 **9**

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 **9**

Need for automata theory - Introduction to formal proof – Finite Automata (FA) – Deterministic Finite Automata (DFA) – Non-deterministic Finite Automata (NFA) – Equivalence between NFA and DFA – Finite Automata with Epsilon transitions – Equivalence of NFA and DFA- Equivalence of NFAs with and without ε-moves- Conversion of NFA into DFA – Minimization of DFAs.

Unit IV Edge Application Development **9**

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 **9**

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.

NUMBER OF THEORY PERIODS:	45

SAMPLE LIST OF EXPERIMENTS

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
11. 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 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,
5. Jiang Zhu, and Sateesh Addepalli (Springer, 1st Edition, 2018),
6. "Edge Computing – The Dawn of Industry 4.0 Revolution" by Satya Prakash Ghrera (CRC Press, 1st Edition, 2020)
7. "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	AI AND CLOUD COMPUTING	L	T	P	C
AD4V67		3	0	0	3

COURSE OBJECTIVES:

1. Understand the fundamentals of basic AI approaches.
2. Learn techniques to identify problem-solving agents
3. To understand the concept of cloud computing.
4. To appreciate the evolution of the cloud from the existing technologies.
5. To have knowledge on the various issues in cloud computing.

UNIT I INTELLIGENT AGENTS **9**

Introduction to AI – Agents and Environments – concept of rationality – nature of environments – structure of agents. Problem solving agents – search algorithms – uninformed search strategies. Constraint satisfaction problems.

UNIT II PROBLEM SOLVING & LOGICAL REASONING **9**

Heuristic search strategies – heuristic functions. Knowledge-based agents – propositional logic – propositional theorem proving –agents based on propositional logic. First-order logic – syntax and semantics –inferences in first-order logic – forward chaining – backward chaining – resolution.

UNIT III CLOUD COMPUTING INTRODUCTION **9**

Introduction to Cloud Computing – Definition of Cloud – Evolution of Cloud Computing – Underlying Principles of Parallel and Distributed Computing – Cloud Characteristics – Elasticity in Cloud – On-demand Provisioning.

UNIT IV CLOUD ENABLING TECHNOLOGIES **9**

Service Oriented Architecture – REST and Systems of Systems – Web Services – Publish-Subscribe Model – Basics of Virtualization – Types of Virtualization – Implementation Levels of Virtualization – Virtualization Structures – Tools and Mechanisms – Virtualization of CPU – Memory – I/O Devices –Virtualization Support and Disaster Recovery.

UNIT V CLOUD ARCHITECTURE, SERVICES AND STORAGE **9**

Layered Cloud Architecture Design – NIST Cloud Computing Reference Architecture – Public, Private and Hybrid Clouds - laaS – PaaS – SaaS – Architectural Design Challenges – Cloud Storage – Storage-as-a-Service – Advantages of Cloud Storage – Cloud Storage Providers – S3.

NUMBER OF THEORY PERIODS: **45**

COURSE OUTCOMES:

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

- CO1 : Learn the intelligent agent frameworks
- CO2 Apply problem solving techniques and CSP techniques
- CO3 Articulate the main concepts, key technologies, strengths and limitations of cloud computing.
- CO4 Learn the key and enabling technologies that help in the development of cloud.
- CO5 Develop the ability to understand and use the architecture of compute and storage cloud, service and delivery models.

TEXT BOOKS:

1. Stuart Russell and Peter Norvig, “Artificial Intelligence – A Modern Approach”, Fourth Edition, Pearson Education, 2021.
2. Kai Hwang, Geoffrey C. Fox, Jack G. Dongarra, "Distributed and Cloud Computing, From Parallel Processing to the Internet of Things", Morgan Kaufmann Publishers, 2012.
3. Rittinghouse, John W., and James F. Ransome, ―Cloud Computing: Implementation, Management and Security‖, CRC Press, 2017.

REFERENCES:

1. Dan W. Patterson, “Introduction to AI and ES”, Pearson Education,2007
2. Kevin Night, Elaine Rich, and Nair B., “Artificial Intelligence”, McGraw Hill, 2008
3. Patrick H. Winston, "Artificial Intelligence", Third Edition, Pearson Education, 2006
4. Deepak Khemani, “Artificial Intelligence”, Tata McGraw Hill Education, 2013.
5. Rajkumar Buyya, Christian Vecchiola, S. ThamaraiSelvi, ―Mastering Cloud Computing‖, Tata Mcgraw Hill, 2013.
6. Toby Velte, Anthony Velte, Robert Elsenpeter, "Cloud Computing - A Practical Approach‖, Tata Mcgraw Hill, 2009.
7. George Reese, "Cloud Application Architectures: Building Applications and Infrastructure in the Cloud: Transactional Systems for EC2 and Beyond (Theory in Practice)‖, O'Reilly, 2009.

Course Code	QUANTUM COMPUTING	L	T	P	C
AD4V68		3	0	0	3

COURSE OBJECTIVES:

1. To introduce the building blocks of Quantum computers and highlight the paradigm change between conventional computing and quantum computing
2. To understand the Quantum state transformations and the algorithms
3. To understand entangled quantum subsystems and properties of entangled states
4. To explore the applications of quantum computing

UNIT I QUANTUM BUILDING BLOCKS **9**

The Quantum Mechanics of Photon Polarization, Single-Qubit Quantum Systems, Quantum State Spaces, Entangled States, Multiple-Qubit Systems, Measurement of Multiple-Qubit States, EPR Paradox and Bell's Theorem, Bloch sphere

UNIT II QUANTUM STATE TRANSFORMATIONS **9**

Unitary Transformations, Quantum Gates, Unitary Transformations as Quantum Circuits, Reversible Classical Computations to Quantum Computations, Language for Quantum Implementations.

UNIT III QUANTUM ALGORITHMS **9**

Computing with Superpositions, Quantum Subroutines, Quantum Fourier Transformations, Shor's Algorithm and Generalizations, Grover's Algorithm and Generalizations

UNIT IV ENTANGLED SUBSYSTEMS AND ROBUST QUANTUM COMPUTATION **9**

Quantum Subsystems, Properties of Entangled States, Quantum Error Correction, Graph states and codes, CSS Codes, Stabilizer Codes, Fault Tolerance and Robust Quantum Computing

UNIT V QUANTUM INFORMATION PROCESSING **9**

Limitations of Quantum Computing, Alternatives to the Circuit Model of Quantum Computation, Quantum Protocols, Building Quantum, Computers, Simulating Quantum Systems, Bell states. Quantum teleportation. Quantum Cryptography, no-cloning theorem

TOTAL PERIODS: 45

COURSE OUTCOMES:

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

CO1 Understand the basic principles of quantum computing.
CO2 Gain knowledge of the fundamental differences between conventional computing and quantum computing.
CO3 Understand several basic quantum computing algorithms.
CO4 Understand the classes of problems that can be expected to be solved well by quantum computers.
CO5 Simulate and analyze the characteristics of Quantum Computing Systems.

TEXT BOOKS:

1 Parag K Lala, Mc Graw Hill Education, Quantum Computing, A Beginners Introduction, First edition (1 November 2020).
2 Michael A. Nielsen, Issac L. Chuang, Quantum Computation and Quantum Information, Tenth Edition, Cambridge University Press, 2010.
3 Chris Bernhardt, The MIT Press; Reprint edition (8 September 2020), Quantum Computing for Everyone.

REFERENCES:

1. John Gribbin, Computing with Quantum Cats: From Colossus to Qubits, 2021
2. William (Chuck) Easttom, Quantum Computing Fundamentals, 2021
3. Parag Lala, Quantum Computing, 2019
4. Eleanor Rieffel and Wolfgang Polak, QUANTUM COMPUTING A Gentle Introduction, 2011
5. Nielsen M. A., Quantum Computation and Quantum Information, Cambridge University Press.2002
6. Benenti G., Casati G. and Strini G., Principles of Quantum Computation and Information, Vol. I: Basic Concepts, Vol II: Basic Tools and Special Topics, World Scientific. 2004
7. Pittenger A. O., An Introduction to Quantum Computing Algorithms 2000

APPENDIX B: OPEN ELECTIVES

Open Electives – I

S. NO.	COURSE CODE	COURSE TITLE	CATEGORY	PERIODS PER WEEK			CREDITS
				L	T	P	
1.	ME46O1	Lean Concepts, Tools and Practices	OEC	3	0	0	3
2.	ME46O9	Machine Learning for Smart Manufacturing	OEC	3	0	0	3
3.	CE46O2	Life cycle Assessment	OEC	3	0	0	3
4.	CE46O5	Environmental Impact Assessment	OEC	3	0	0	3
5.	CS46O5	Emotional Intelligence	OEC	3	0	0	3

Open Electives – II

S. NO.	COURSE CODE	COURSE TITLE	CATEGORY	PERIODS PER WEEK			CREDITS
				L	T	P	
1.	ME47O1	Technical Writing	OEC	3	0	0	3
2.	EE47O4	Drinking Water Supply and Treatment	OEC	3	0	0	3
3.	CE47O4	Geographic Information System	OEC	3	0	0	3
4.	ME47O5	Renewable Energy Technologies	OEC	3	0	0	3
5.	CS47O3	Green Computing	OEC	3	0	0	3

Open Electives – I

Course Code	LEAN CONCEPTS, TOOLS AND PRACTICES	L	T	P	C
ME4601		3	0	0	3

COURSE OBJECTIVES:

- To impart knowledge about the basics of lean principles, tools and techniques, and implementation in the construction industry.

UNIT I Introduction 9

Introduction and overview of the construction project management - Review of Project Management & Productivity Measurement Systems - Productivity in Construction - Daily Progress Report-The state of the industry with respect to its management practices -construction project phases - The problems with current construction management techniques.

UNIT IILean Management 9

Introduction to lean management - Toyota's management principle-Evolution of lean in construction industry - Production theories in construction –Lean construction value - Value in construction - Target value design - Lean project delivery system- Forms of waste in construction industry - Waste Elimination.

UNIT III Core Concepts in Lean 9

Concepts in lean thinking – Principles of lean construction – Variability and its impact – Traditional construction and lean construction – Traditional project delivery - Lean construction and workflow reliability – Work structuring – Production control.

UNIT IV Lean Tools and Techniques 9

Value Stream Mapping – Work sampling – Last planner system – Flow and pull based production – Last Planner System – Look ahead schedule – constraint analysis – weekly planning meeting-Daily Huddles – Root cause analysis – Continuous improvement – Just in time.

UNIT VLean Implementation in Construction Industry 9

Lean construction implementation- Enabling lean through information technology - Lean in design - Design Structure - BIM (Building Information Modelling) - IPD (Integrated Project Delivery) – Sustainability through lean construction approach.

TOTAL: 45 PERIODS

COURSE OUTCOMES:

CO1: Explains the contemporary management techniques and the issues in present scenario.
CO2: Apply the basics of lean management principles and their evolution from manufacturing industry to construction industry.
CO3: Develops a better understanding of core concepts of lean construction tools and techniques and their importance in achieving better productivity.
CO4: Apply lean techniques to achieve sustainability in construction projects.
CO5: Apply lean construction techniques in design and modeling.

REFERENCES:

1. Corfe, C. and Clip, B., Implementing lean in construction: Lean and the sustainability agenda, CIRIA, 2013.
2. Shang Gao and Sui Pheng Low, Lean Construction Management: The Toyota Way, Springer, 2014.

3. Dave, B., Koskela, L., Kiviniemi, A., Owen, R., andTzortzopoulos, P., Implementing lean in construction: Lean construction and BIM, CIRIA, 2013.
4. Ballard, G., Tommelein, I., Koskela, L. and Howell, G., Lean construction tools and techniques, 2002.
5. Salem, O., Solomon, J., Genaidy, A. and Luegring, M., Site implementation and Assessment of Lean Construction Techniques, Lean Construction Journal, 2005.

Course Code	MACHINE LEARNING FOR SMART MANUFACTURING	L	T	P	C
ME46O9		3	0	0	3

COURSE OBJECTIVES:
Impart knowledge of smart manufacturing for industry 4.0 for making student innovative.

UNIT I Industry 4.0 **9**
Concept, Globalization and emerging issues, The Fourth Revolution, LEAN manufacturing, Smart and connected business perspectives, Smart factories.

UNIT II Automation **9**
Programable Logic Controller (PLC) and its Programming software, Communication of different devices with PLC, Sensor, Smart Sensor, HMI design, Cyber Physical System – key components, ISA-95 architecture, CPS-5C architecture, Concept of Digit Twin.

UNIT III Communication **9**
Protocols – MQTT, OPC UA, EtherNet/IP, Profinet, EtherCAT, etc; MQTT – History, MQTT broker, Message types, Quality of Service (QoS), Application; OPC UA – History, Specification, Client, Server, Programming with – Free and open-source software, Propriety software; Augmented Reality.

UNIT IV IoT Platform **9**
Data Modelling, IoT platforms – Thing, basic functionalities, Abstract definition of Thing, Networks, etc; IoT Gateway, Machine interfaces – Cloud-based Mosquitto brokers, Programming with – Free and open-source software, Propriety software.

UNIT V Machine Learning Foundation **9**
Learning algorithms – Supervised, Unsupervised, Self-learning, Feature learning, etc. Models – Artificial Neural Networks, Decision trees, Regression analysis, Genetic algorithms, etc.; Programming with – Free and open-source software, Propriety software.

TOTAL: 45 PERIODS

COURSE OUTCOMES:
On successful completion of this course, the student will be able to
CO1: Introduce concept of Industry 4.0 for Smart Manufacturing.
CO2: Understand use various hardware used in Smart Manufacturing.
CO3: Understand need of various communication protocols. hardware and software, IoT Layers and their relative importance.
CO4: Understand could-computing IoT platform for Smart Manufacturing.
CO5: Understand machine learning to make smart factories.
CO6: Understand application of hardware, communication protocol, IOT platform, machine learning etc. to implement IoT for smart manufacturing for the need of Industry 4.0.

REFERENCES:

1. Christoph Jan Bartodziej, "The Concept Industry 4.0 – An Empirical Analysis of Technologies and Application in Production Logistics", Spinger Gabler, 2015 2.
2. Alasdair Gilchrist, "Industry 4.0 – The Industrial Internet of Thigs", Springer Link, 2016 3.
3. Daniel Minoli, "Building the Internet of Things with IPv6 and MIPv6: The Evolving World of M2M Communications", ISBN: 978-1-118-47347-4, Willy Publications. 4.
4. Michahelles, "Architecting the Internet of Things", ISBN 978-3- 642-19156-5 e-ISBN 978-3-642-19157-2, Springer. 5.

5. Hakima Chaouchi, “The Internet of Things Connecting Objects to the Web” ISBN : 978-1-84821-140-7, Willy Publications. 6.
6. Olivier Hersent, David Boswarthick, Omar Elloumi, “The Internet of Things: Key Applications and Protocols”, ISBN: 978-1-119-99435-0, 2nd Edition, Willy Publications 7.
7. W. Botton, “Programmable Logic Controllers”, Fourth Edition, Elsevier, 2006
8. P. Juahs, K. Molnar, “Key Components of the Architecture of Cyber-physical manufacturing systems”, International Scientific Journal “Industry 4.0”, 2017, issue 5, 205- 207
9. Jen-Ruey Jiang, “An improved cyber-physical systems architecture for Industry 4.0 smart factories”, Advances in Mechanical Engineering, 2018, Vol. 10(6) 1-15

Course Code	Life cycle assessment	L	T	P	C
CE4602		3	0	0	3

COURSE OBJECTIVES:

The main objectives of this course are to:

1. To impart knowledge and skills on the concept and methodology of Life Cycle Assessment as per international standards and its potential applications
2. To develop sustainable products and promote sustainable consumption.
3. Understanding of the principles, methodologies, and techniques involved in Life Cycle Assessment (LCA).
4. Develop the ability to identify, quantify, and assess the environmental impacts associated with various stages of a product or system's life cycle.
5. Learn how to interpret LCA results and communicate findings effectively, including understanding the significance of different impact categories.

Course Description

This course provides an in-depth exploration of the principles, methodologies, and applications of LCA, equipping students with the knowledge and skills necessary to conduct comprehensive environmental assessments and make informed decisions to promote sustainability.

Prerequisites

- Familiarity with the concept of life cycle thinking and systems theory.
- Understanding of fundamental environmental science concepts such as ecology, environmental impacts, and sustainability principles provides a foundational knowledge base for studying LCA.

UNIT I Life Cycle Thinking and Life Cycle Management 9

Introduction to Life Cycle Thinking – Industrial ecology – Life cycle management (LCM) and Stakeholder Expectations - LCM drivers and issues - materials flow analysis - Life cycle of Products and services- International organizations and networks - History and definition of LCA - analytical tools for product and service systems –––Value creation along the life cycle– technical characteristics – applications – limitations

UNIT II LCA Goal, Scope and Inventory 9

ISO 14040 framework for LCA - Life cycle goal and scope definition - function, functional unit and reference flow System boundaries, data categories, inputs and outputs, data quality, critical review and other procedural aspects - Inventory Analysis: Raw Material Extraction and Processing , Manufacturing and Production , Product Use and Consumption , End-of-life Management, Transportation and Distribution - Dealing with Allocation Issues - Solutions to the multi functionality problem - Flow diagram - Format and data categories - Attributional versus consequential LCI – LCA software and database - Data quality - Data collection and relating data to unit processes – Data validation - Cut-off and data estimation .

UNIT III Life Cycle Impact Analysis and Interpretation 9

Characterization factors and principle of characterization - Selection of impact categories, category indicators and characterization models – Classification -Characterization - Optional elements - normalization , grouping, weighting ,data quality analysis - Characterization models – Impact assessment Case studies -Simplified/streamlined Life Cycle Assessments – procedural

approaches, numerical approaches - Examples of numerical approaches - contribution analysis, perturbation analysis, uncertainty - analysis, comparative analysis, key issue analysis – Treatment of uncertainties - Elements in uncertainty handling - Sensitivity of LCA results – Sustainability analysis - Extending LCA - economic dimension, social dimension - Life cycle costing – Eco efficiency - Combining LCA and LCC – Case studies

UNIT IV Design for Environment and Ecolabelling 9

Sustainable consumption – Eco-efficiency - green consumerism - product stewardship and green engineering - Extended producer responsibility – Design For Environment Strategies, Practices, Guidelines, Methods, And Tools .Eco design strategies –Design for Disassembly - Dematerialization, re materialization, trans materialization – Green procurement and green distribution - Analysis framework for reuse and recycling – Typical constraints on reuse and recycling - Communication of Life Cycle Information - Indian eco mark scheme – Environmental product declarations – Environmental marketing

UNIT V LCA Softwares and Case Studies 9

LCA Softwares - LCA Software Demo: SimaPro, GREET, BEES, CMU EIO,GABI - Advances in LCA: Hybrid LCA, Thermodynamic LCA - LCA case studies on Product Design, Product Improvement, Product Comparison and Policy development.

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:

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

CO1: Explain the various functional elements of Life Cycle Analysis and Design for Environment
CO2: Apply the knowledge of science and engineering fundamentals to characterize the environmental interactions of products and services
CO3: Design of engineering systems taking into account the material flow and pollutant interactions between engineering decisions and the environment
CO4: Select appropriate LCA tools to support product/process design and decision making, taking into account the impact of the solutions in a sustainability context
CO5: Evaluate generative models using appropriate metrics and critically analyse the ethical implications, privacy concerns, and societal impact of generative AI technologies.

TEXT BOOKS:

1. Ralph Horne, Tim Grant, Karli Verghese, Life Cycle Assessment: Principles, Practice and Prospects, Csiro Publishing, 2009.

REFERENCES:

1. ISO 14040-2016-Environmental management - Life cycle assessment - Principles and framework, International Organization for Standardization, 2016.
2. ISO/TR 14047:2003, Environmental management - Life cycle impact assessment - Examples of application of ISO 14042, International Organization for Standardization, 2007.

3. International Organization for Standardization: ISO TR 14062 Environmental management Integrating environmental aspects into product design and development, 2002.
4. European Commission - Joint Research Centre - Institute for Environment and Sustainability: International Reference Life Cycle Data System (ILCD) Handbook – General guide for Life Cycle Assessment - Detailed guidance. Luxembourg. European Union; 2010.
5. Catherine Benoit, UQAM/CIRAIG, and Bernard Mazijn, Guidelines for Social Life Cycle Assessment of Products, United Nations Environment Programme,2009

TOTAL: 45 PERIODS

YouTube Resources:

1. **Sustainable Minds -** Sustainable Minds is a provider of cloud-based software and services for sustainable product development.
2. **Life Cycle Assessment – SimaPro** - SimaPro is one of the leading software tools for conducting Life Cycle Assessments.
3. **European Commission – JRC** - The Joint Research Centre (JRC) of the European Commission produces research and technical reports on a wide range of topics, including environmental sustainability and Life Cycle Assessment.
4. **Life Cycle Initiative** - The Life Cycle Initiative, hosted by the United Nations Environment Programme (UNEP), promotes the use of life cycle approaches for sustainable development.
5. **LCA Learning** - This channel offers various educational videos on Life Cycle Assessment, including tutorials, case studies, and discussions on LCA methodology and applications:

Course Code	Environmental Impact Assessment	L	T	P	C
CE4605		3	0	0	3

COURSE OBJECTIVES:

The main objectives of this course are to:

1. To expose the students to the need, methodology, documentation and usefulness of environmental impact assessment and to develop the skill to prepare environmental management plan.
2. To participate in the performance of an environmental assessment process (EIA or SEA), given the disciplinary knowledge and skills in natural sciences and engineering the student have achieved in other courses.

Course Description

This course provides an introduction to the theory and practical applications of environmental impact. Students will learn the fundamental concepts and techniques related to environmental impact and gain hands-on experience with creating and using environmental impact assessment.

Prerequisites

- Basic knowledge of environmental impact assessment.
- Familiarity with an environmental issue

UNIT I Introduction **9**

Historical development of Environmental Impact Assessment (EIA). Environmental Clearance- EIA in project cycle. legal and regulatory aspects in India – types and limitations of EIA –EIA process screening – scoping - terms of reference in EIA- setting – analysis – mitigation. Cross sectoral issues –public hearing in EIA- EIA consultant accreditation.

UNIT II Impact Identification And Prediction **9**

Matrices – networks – checklists – cost benefit analysis – analysis of alternatives – expert systems in EIA. Prediction tools for EIA – mathematical modelling for impact prediction – assessment of impacts – air – water – soil – noise – biological –– cumulative impact assessment

UNIT III Socio-Economic Impact Assessment **9**

Socio-economic impact assessment - relationship between social impacts and change in community and institutional arrangements. factors and methodologies- individual and family level impacts. communities in transition-rehabilitation

UNIT IV EIA Documentation And Environmental Management Plan **9**

Environmental management plan - preparation, implementation and review – mitigation and rehabilitation plans – policy and guidelines for planning and monitoring programmes – post project audit – documentation of EIA findings – ethical and quality aspects of environmental impact assessment

UNIT V Case Studies **9**

Mining, power plants, cement plants, highways, petroleum refining industry, storage & handling of hazardous chemicals, common hazardous waste facilities, CETPs, CMSWMF, building and construction projects.

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:

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

CO1: Carry out scoping and screening of developmental projects for environmental and social Assessments
CO2: Explain different methodologies for environmental impact prediction and assessment.
CO3: Assessing socio-economic investigation of the environment as a project.
CO4: Plan environmental impact assessments and environmental management plans.
CO5: Knowledge to prepare environmental impact assessment reports for various projects.

TEXT BOOKS:

1. Canter, L.W., "Environmental Impact Assessment", McGraw Hill, New York. 1996.
2. Lawrence, D.P., "Environmental Impact Assessment – Practical solutions to recurrent problems",Wiley-Interscience, New Jersey. 2003.

REFERENCES:

1. World Bank –Source book on EIA
2. Cutter, S.L., "Environmental Risk and Hazards", Prentice-Hall of India Pvt. Ltd., New Delhi, 1999.
3. Kolluru Rao, Bartell Steven, Pitblado R and Stricoff "Risk Assessment and Management
4. Handbook", McGraw Hill Inc., New York,1996.
5. K.V.Raghavan and A A. Khan, "Methodologies in Hazard Identification and Risk Assessment",Manual by CLRI, 1990.

TOTAL :45 PERIODS

YouTube Resources:

1. **IIT Roorkee**- EIA is basically a tool used to assess the positive and negative environmental, economic and social impacts of a project.
2. **Science Sauce** - Explaining tricky Science concepts in as little time as possible. All content created by Alex Nixon..
3. **Border Archaeology** - They are mandated by The Town and Planning Regulations 2011 for projects falling under Schedule 1 and may also be sought by a local planning authority for Schedule 2-type projects following project screening and scoping..
4. **Sustainable Technology Solutions** - The relevance of EIA lies in its ability to ensure that projects are designed and implemented in a way that minimizes environmental harm and promotes sustainable development.
5. **Sigma Earth** - Climate change refers to long-term shifts in temperatures and weather patterns. Human activities have been the main driver of climate change, primarily due to the burning of fossil fuels like coal, oil and gas..

Course Code	EMOTIONAL INTELLIGENCE	L	T	P	C
CS4606		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
2. Life: A Scientific Inquiry. Psychology Press: Philadelphia, PA.
3. Daliph Singh (2001) Emotional Intelligence At Work: A Professional Guide. Response Books: New Delhi
4. Daniel Goleman, (1996) Emotonal Intelligence: Why It can Matter More Than IQ.
5. Bantam Books: NewYork.
6. Doty, G. (2001). Fostering Emotional Intelligence in K-8 Students. Corwin Press:
7. Thousand Oaks, CA.
8. Oatley, K and Jenkins, J (2000) Understanding Emotions. Malden MA: Blackwell

Course Code	TECHNICAL WRITING	L	T	P	C
ME47O1		3	0	0	3

COURSE OBJECTIVES:

- To understand the present complex information in a clear and easily understandable way to the target audience, which involves breaking down intricate concepts into simpler terms and providing clear explanations.

UNIT I Introduction to Technical Writing 9

Characteristics of Technical Writing - Rhetorical awareness - Ethics - Steps in the technical writing process- Prewriting for technical documents-Understanding audience and purpose, Primary and secondary research - Surveys and interviews - Research methods

UNIT II Components of Technical Documents 9

Introductions – Abstracts – Definitions – Titles and headings - Effective visual design – Summaries – Technical descriptions – conclusions

UNIT III Types of Technical Report 9

Formal Technical Reports - Progress and research reports - Incidence reports - Feasibility reports - Evaluation reports – Analytical and informational reports - Executive summaries.

UNIT IV Language 9

Style – Accuracy – Brevity – Clarity – Tone – Vocabulary – Formal and impersonal language – Structure of the report - Plagiarism.

UNIT V Writing Proposals 9

Nature and significance –Types of proposals - Persuasive elements - Request for proposals – Structure and parts of a proposal

TOTAL: 45 PERIODS

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

CO1: Acquire a working knowledge of writing strategies, formats and templates of professional writing.

CO2: Analyse communication-related problems of technical documents from number of genres.

CO3: Use visuals to communicate a large amount of information quickly and efficiently

CO4: Enhance writing skills to produce effective reports confidently

TEXT BOOKS:

1. Daniel G. Riordan, Steven E. Pauley, Biztantra: Technical Report Writing Today, 8th Edition (2004).
2. Rizvi M Ashraf, (2005). Effective Technical Communication. McGraw Hill Education (India) Pvt. Ltd. New Delhi.
3. Alred, G. (2011). Handbook of Technical Writing (10th ed.). New York: St Martin's. (OPTIONAL)

REFERENCES:

1. M. Frank. Writing as thinking: A guided process approach, Englewood Cliffs, Prentice Hall Reagents.
2. R. Quirk, S. Greenbaum, G. Leech and J. Svartik: A comprehensive grammar of the English language, Longman, London.
3. Daniel G. Riordan & Steven A. Panley: "Technical Report Writing Today" - Biztaantra.

Course Code	Drinking water supply and treatment	L	T	P	C
EE4704		3	0	0	3

COURSE OBJECTIVES:

The main objectives of this course are to:

- To equip the students with the principles and design of water treatment units and distribution system.
- To understand the sources of water and their characteristics.
- To gain knowledge of conveyance systems, including pipes, conduits, and pumps.
- To understand the principles and processes of water treatment, including coagulation, filtration, and disinfection.
- To explore advanced water treatment technologies such as desalination, membrane systems, and ion exchange.

Course Description

This course provides students with an in-depth understanding of water supply systems, including the planning, design, and operation of water sources, conveyance systems, treatment plants, and distribution networks. Students will learn about the characteristics of different water sources, the design of intake structures and transmission mains, principles and processes of water treatment, advanced water treatment technologies, and the design and operation of water distribution systems.

Prerequisites

- Basic knowledge of fluid mechanics and hydraulics.
- Understanding of environmental science and water quality parameters.
- Familiarity with engineering design principles.

UNIT I Sources of Water **9**

Public water supply system – Planning, Objectives, Design period, Population forecasting; Water demand – Sources of water and their characteristics, Surface and Groundwater – Impounding Reservoir – Development and selection of source – Source Water quality – Characterization – Significance – Drinking Water quality standards.

UNIT II Conveyance from the Source **9**

Water supply–in take structures–Functions; Pipes and conduits for water –Pipe materials – Hydraulics of flow in pipes –Transmission main design – Laying, jointing and testing of pipes – appurtenances – Types and capacity of pumps – Selection of pumps and pipe materials.

UNIT III Water Treatment **9**

Objectives–Unit operations and processes–Principles, functions, and design of water treatment plant units, aerators of flash mixers, Coagulation and flocculation's and filters-Disinfection–Construction, Operation and Maintenance aspects.

UNIT IV Advanced Water Treatment **9**

Water softening – Desalination- R.O. Plant – demineralization –Adsorption – Ion exchange–Membrane Systems – Iron and Manganese removal – Defluorination – Construction and Operation and Maintenance aspects

UNIT V Water Distribution and Supply **9**

Requirements of water distribution – Components – Selection of pipe material – Service reservoirs – Functions – Network design – Economics – Computer applications – Appurtenances– Leak detection Principles of design of water supply in buildings – House service connection– Fixtures and fittings, systems of plumbing and types of plumbing

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:

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

CO1 An understanding of water quality criteria and standards, and their relation to public health
CO2 The ability to design the water conveyance system
CO3 The knowledge in various uni tope rations and processes in water treatment
CO4 An ability to understand the various systems for advanced water treatment

TEXT BOOKS:

1. Garg. S.K., "Water Supply Engineering", Khanna Publishers, Delhi, September 2008.
2. Punmia B.C,ArunK.Jain,Ashok K.Jain, " Water supply Engineering" Lakshmi publication private limited, New Delhi, 2016.
3 . Rangwala"WaterSupplyandSanitaryEngineering",February2022
1. Birdie.G.S., "Water Supply and Sanitary Engineering", Dhanpat Rai and sons, 2018.

REFERENCES:

1. Fair.G.M.,Geyer.J.C.,"WaterSupplyandWastewaterDisposal",JohnWileyand Sons, 1954.
2. Babbit.H.E,andDonald.J.J,"WaterSupplyEngineering",McGrawHillbookCo, 1984.
3. Steel. E.W.et al., "Water Supply Engineering" , Mc Graw Hill International book Co, 1984.
4. Duggal.K.N.,"ElememtsofpublicHealthEngineering",S.ChandandCompanyLtd,NewDelhi,1998.

YouTube Resources:

1. https://youtu.be/yZwfcMSDBHs?si=h-va7awNWu862fMB
2. https://youtu.be/ZQKpu-obzlU?si=0DUbNWO0rw7RPq_q
3. https://youtu.be/u4k2XY-fJJY?si=5EQUC2t6NuJlFhEx
4. https://youtu.be/Ki8LmnPt6qE?si=5X2oJ-3vltWlT35I
5. https://youtu.be/iyVdiQonEA0?si=9OoaiwtiHGKablZj

TOTAL: 45 PERIODS

Course Code	Geographic Information System	L	T	P	C
CE4704		3	0	0	3

COURSE OBJECTIVES:

The main objectives of this course are to:

- Introduce concepts of Cartography and GIS
- Expose the process of map making and production
- Introduce GIS data structures, data input and data presentation Apply principles of sustainability and resilience to civil infrastructure projects.

Course Description

This course provides an introduction on concepts and principles of mapping the surface components, geographical terrain projection, Data inputs and Topology.

Prerequisites

- Basic knowledge in Engineering and Geographic surveys.
- Familiarity with computer software for engineering analysis and design, such as AutoCAD, QGIS (Geographic Information Systems).

UNIT I Elements Of Cartography **9**

Definition of Cartography – Maps – Functions – Uses and Types of Maps – Map Scales and Contents – Map Projections – Shape, Distance, Area and Direction Properties – Perspective and mathematical Projections – Indian Maps and Projections – Map Co-ordinate System – UTM and UPS References.

UNIT II Map Design and Production **9**

Elements of a Map – Map Layout Principles – Map Design Fundamentals – Symbols and Conventional Signs – Graded and Ungraded Symbols – Color Theory – Colours and Patterns in Symbolization – Map Lettering – Map Production – Map Printing – Colours and Visualization – Map Reproduction – Map Generalization – Geometric Transformations – Bilinear and Affine Transformations.

UNIT III Fundamentals Of GIS **9**

Introduction to GIS – Definitions – History of GIS – Components of a GIS – Hardware, Software, Data, People, Methods – Introduction to data quality – Types of data – Spatial, Attribute data – types of attributes – scales/levels of measurements – spatial data models – Raster Data Structures – Raster Data Compression – Vector Data Structures – Raster Vs Vector Models – TIN and GRID data models.

UNIT IV Data Input And Topology **9**

Image Scanner – Raster Data Input – Raster Data File Formats – Georeferencing – Vector Data Input – Digitizer– Datum Projection and Reprojection – Coordinate Transformation – Topology - Adjacency, Connectivity and containment – Topological Consistency – Non topological file formats – Attribute Data Linking – Linking External Databases – GPS Data Integration – Raster to Vector and Vector to Raster Conversion.

UNIT V Data Quality And Output **9**

Assessment of Data Quality – Basic Aspects – Completeness, Logical Consistency, Positional Accuracy, Temporal Accuracy, Thematic Accuracy and Lineage – Metadata – GIS Standards –

Interoperability – OGC – Spatial Data Infrastructure – Data Output – Map Compilation – Chart / Graphs.

Course Format

Lectures and discussions, Hands-on 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:

OUTCOMES: Upon completion of the course, the student is expected to:

CO1: Be familiar with appropriate map projection and co-ordinate system for production of Maps and shall able to compile and design maps for their required purpose.
CO2: Be familiar with co-ordinate and Datum transformations
CO3: Understand the basic concepts and components of GIS, the techniques used for storage of spatial data and data compression
CO4: Understand the concepts of spatial data quality and data standard
CO5: Understand the concept of spatial data inputs

TEXT BOOKS:

1. Arthur H. Robinson et al, "Elements of Cartography", 7th Edition, Wiley, 2002.
2. Kang – Tsung Chang, "Introduction to Geographic Information Systems", McGraw Hill Publishing, Fourth Edition, 2017.
3. Ian Heywood, Sarah Cornelius, Steve Carver, Srinivasa Raju, "An Introduction to Geographical Information Systems, Pearson Education, Fourth Edition, 2011.

REFERENCES:

1. John Campbell, "Introductory Cartography", Wm. C.BrownPublishers,3rd Edition,2004
2. Chor Pang LO, Albert K. W. Yeung, "Concepts and Techniques of Geographic Information Systems", Pearson Education, 2nd Edition, November 2016. ISBN: 9789332581883

TOTAL:45 PERIODS

YouTube Resources:

1. **Esri** – Esri is a leading provider of GIS software and solutions. Their YouTube channel features tutorials, webinars, case studies, and product demonstrations related to ArcGIS, one of the most widely used GIS platforms globally.
2. **GIS Tutorial** - This channel offers tutorials and guides for learning GIS concepts, software usage (especially ArcGIS), and practical applications in various fields such as urban planning, environmental science, and engineering.
3. **GeoSpatial Training** - GeoSpatial Training provides video tutorials and webinars covering a wide range of GIS topics, including software tutorials, data analysis techniques, and industry-specific applications.
4. **QGIS Tutorials and Tips** - QGIS is an open-source GIS software alternative to ArcGIS. This channel offers tutorials, tips, and tricks for using QGIS effectively, covering topics from basic to advanced functionalities.
5. **MapScaping** - MapScaping produces informative videos on GIS, cartography, remote sensing, and geospatial technologies. Their content includes tutorials, interviews with GIS professionals, and discussions on emerging trends in the field.

Course Code	RENEWABLE ENERGY TECHNOLOGIES	L	T	P	C
ME4705		3	0	0	3

COURSE OBJECTIVES:

- To know the Indian and global energy scenario
- To learn the various solar energy technologies and its applications.
- To educate the various wind energy technologies.
- To explore the various bio-energy technologies.
- To study the ocean and geothermal technologies.

UNIT I Energy Scenario **9**

Indian energy scenario in various sectors – domestic, industrial, commercial, agriculture, transportation and others – Present conventional energy status – Present renewable energy status. Potential of various renewable energy sources-Global energy status-Per capita energy consumption - Future energy plans.

UNIT II Solar Energy **9**

Solar radiation – Measurements of solar radiation and sunshine – Solar spectrum - Solar thermal collectors – Flat plate and concentrating collectors – Solar thermal applications – Solar thermal energy storage – Fundamentals of solar photo voltaic conversion – Solar cells – Solar PV Systems – Solar PV applications

UNIT III Wind Energy **9**

Wind data and energy estimation – Betz limit - Site selection for windfarms – characteristics - Wind resource assessment - Horizontal axis wind turbine – components - Vertical axis wind turbine – Wind turbine generators and its performance – Hybrid systems – Environmental issues – Applications.

UNIT IV Bio-Energy **9**

io resources – Biomass direct combustion – thermochemical conversion - biochemical conversionmechanical conversion - Biomass gasifier - Types of biomass gasifiers - Cogeneration –- Carbonisation – Pyrolysis - Biogas plants – Digesters –Biodiesel production – Ethanol production – Applications

UNIT V Ocean and geothermal energy **9**

Small hydro - Tidal energy – Wave energy – Open and closed OTEC Cycles – Limitations – Geothermal energy – Geothermal energy sources - Types of geothermal power plants – Applications - Environmental impact

TOTAL: 45 PERIODS

COURSE OUTCOMES:

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

CO1: Discuss the Indian and global energy scenario.
CO2: Describe the various solar energy technologies and its applications.
CO3: Explain the various wind energy technologies.
CO4: Explore the various bio-energy technologies.
CO5: Discuss the ocean and geothermal technologies.

TEXT BOOKS:

1. Fundamentals and Applications of Renewable Energy | Indian Edition, by Mehmet Kanoglu, Yunus A. Cengel, John M. Cimbala, cGraw Hill; First edition (10 December 2020), ISBN-10 : 9390385636
2. Renewable Energy Sources and Emerging Technologies, by Kothari, Prentice Hall India Learning Private Limited; 2nd edition (1 January 2011), ISBN-10 : 812034470

REFERENCES:

1. Godfrey Boyle, "Renewable Energy, Power for a Sustainable Future", Oxford University Press, U.K., 2012.
2. Rai.G.D., "Non-Conventional Energy Sources", Khanna Publishers, New Delhi, 2014.
3. Sukhatme.S.P., "Solar Energy: Principles of Thermal Collection and Storage", Tata McGraw Hill Publishing Company Ltd., New Delhi, 2009.
4. Tiwari G.N., "Solar Energy – Fundamentals Design, Modelling and applications", Alpha Science Intl Ltd, 2015.
5. Twidell, J.W. & Weir A., "Renewable Energy Resources", EFNSpon Ltd., UK, 2015

Course Code	GREEN COMPUTING	L	T	P	C
CS4703		3	0	0	3

CORE 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)

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 for Literature	III	2	0	0	0
3.	MC4303	Film Appreciation	III	2	0	0	0
4.	MC4304	Disaster Management	III	2	0	0	0
5.	MC4305	Design Thinking	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	Environmental Science and Sustainability	IV	2	0	0	0
2.	MC4402	Well Being with Traditional practices (Yogam Ayurveda and Siddha)	IV	2	0	0	0
3.	MC4403	History of Science and Technology in India	IV	2	0	0	0
4.	MC4404	Political and Economic Thought for a Humane Society	IV	2	0	0	0
5.	MC4405	State, Nation building and Politics in India	IV	2	0	0	0
6.	MC4406	Industrial Safety	IV	2	0	0	0
Total Credits				**0**			

MANDATORY COURSE I

Course Code	INTRODUCTION TO WOMEN AND GENDER STUDIES	L	T	P	C
MC4301		2	0	0	0

COURSE OBJECTIVES:

- Understand the difference between sex and gender, including masculinity and femininity, shaped by societal norms.
- Critique patriarchal systems and hierarchies, advocating for the deconstruction of binary views and recognition of gender diversity.
- Engage with various feminist frameworks to analyze and address gender inequality.
- Trace the development and impact of feminist movements globally, nationally, and locally.
- Analyze language, media, and narratives to understand and challenge gender norms and stereotypes.

UNIT I CONCEPTS 2

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 2

Liberal, Marxist, Socialist, Radical, Psychoanalytic, postmodernist, ecofeminist.

UNIT III WOMEN'S MOVEMENTS: GLOBAL, NATIONAL AND LOCAL 2

Rise of Feminism in Europe and America, Women's Movement in India.

UNIT IV GENDER AND LANGUAGE 2

Linguistic Forms and Gender, Gender and narratives.

UNIT V GENDER AND REPRESENTATION 2

Advertising and popular visual media, Gender and Representation in Alternative Media, Gender and social media.

TOTAL:10 PERIODS

Course Outcomes:

Upon Successful Completion of the course the students will be able to

CO1: Mastery of key gender studies concepts, including sex vs. gender, patriarchy, and power dynamics.

CO2: Ability to critically evaluate societal norms and gender roles, recognizing and deconstructing stereotypes.

CO3: Application of diverse feminist theories to address gender inequality on local, national, and global scales.

CO4: Understanding of the historical progression and impact of feminist movements on social attitudes and policies.

CO5: Development of effective communication skills to advocate for gender equality, challenging norms in language, media, and society architecture and how different ARM processors work.

Text Books & 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	ELEMENTS OF LITERATURE	L	T	P	C
MC4302		2	0	0	0

COURSE OBJECTIVES:

- 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 RELEVANCE OF LITERATURE 2

Enhances Reading, thinking, discussing and writing skills, Develops finer sensibility for better human relationship, Increases understanding of the problem of humanity without bias, Providing space to reconcile and get a cathartic effect.

UNIT II ELEMENTS OF FICTION 2

Fiction, fact and literary truth, Fictional modes and patterns, Plot character and perspective.

UNIT III ELEMENTS OF POETRY 2

Emotions and imaginations, Figurative language, Simile, metaphor, conceit, symbol, pun and irony, Personification and animation, Rhetoric and trend.

UNIT IV ELEMENTS OF DRAMA 2

Drama as representational art, Content mode and elements, Theatrical performance, Drama as narration, mediation and persuasion, Features of tragedy, comedy and satire.

UNIT V TUTORIALS 2

The students will write a term paper to show their understanding of a particular piece of literature

TOTAL:10 PERIODS

Course Outcomes:

Upon Successful Completion of the course the students will be able to

Students will be able to understand the relevance of literature in human life and appreciate its aspects in developing finer sensibilities

Text Books:

1. An Introduction to the Study of English Literature, W.H. Hudson, Atlantic, 2007.
2. An Introduction to Literary Studies, Mario Klarer, Routledge, 2013.

References:

1. The Experience of Poetry, Graham Mode, Open college of Arts with Open Unv Press,1991.
2. The Elements of Fiction: A Survey, Ulf Wolf (ed), Wolfstuff, 2114.

Course Code	FILM APPRECIATION	L	T	P	C
MC4303		2	0	0	0

COURSE OBJECTIVES:

- Understand filmmaking components, including equipment and storytelling.
- Trace film language evolution, focusing on significant milestones.
- Explore film theories (realist, auteurist, psychoanalytic, feminist).
- Analyze representative films from diverse cultural contexts.
- Examine Indian cinema's development, from early era to regional diversity..

UNIT I THE COMPONENT OF FILMS 2

The material and equipment, The story, screenplay and script, The actors, crew members, and the director, The process of film making, Structure of a film

UNIT II EVOLUTION OF FILM LANGUAGE 2

Film language, form, movement etc, Early cinema, Silent film (Particularly French), The emergence of feature films: Birth of a Nation, Talkies

UNIT III FILM THEORIES AND CRITICISM/APPRECIATION 2

Realist theory: Auteurists, Psychoanalytic, Ideological, Feminists, How to read films? Film Criticism / Appreciation

UNIT IV DEVELOPMENT OF FILMS 2

Representative Soviet films, Representative Japanese films, Representative Italian films, Representative Hollywood film and the studio system.

UNIT V INDIAN FILMS 2

The early era, The important films made by the directors, The regional films, The documentaries in India

TOTAL:10 PERIODS

Course Outcomes:

Upon Successful Completion of the course the students will be able to

CO1: Gain a comprehensive understanding of filmmaking components and techniques

CO2: Explore the historical evolution of film language and major milestones.

CO3: Develop critical analysis skills through the study of various film theories.

CO4: Broaden cultural awareness by analyzing representative films from diverse contexts.

CO5: Appreciate the development and diversity of Indian cinema

Text Books:

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 MANAGEMENT	L	T	P	C
MC4304		2	0	0	0

COURSE OBJECTIVES:

- To impart knowledge on concepts related to disaster, disaster risk reduction, disaster
- management
- To acquaint with the skills for planning and organizing disaster response

UNIT I HAZRADS, VULNERABILITY AND DISASTER RISKS 2

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) 2

Sendai Framework for Disaster Risk Reduction, Disaster cycle - Phases, Culture of safety, prevention, mitigation and preparedness community Based DRR, Structural- nonstructural 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 2

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 2

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 2

Discussion on selected case studies to analyze 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 -

TOTAL:10 PERIODS

Course Outcomes:

Upon Successful Completion of the 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

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

References:

1. Govt. of India: Disaster Management Act, Government of India, New Delhi, 2005.
2. Government of India, National Disaster Management Policy, 2009.

Course Code	DESIGN THINKING	L	T	P	C
MC4305		2	0	0	0

COURSE OBJECTIVES:

The main objectives of this course are to:

- Introduce students to the fundamental concepts and principles of design thinking.
- Develop students' ability to empathize with users and identify their needs.
- Equip students with skills for problem-solving and generating innovative solutions.
- Foster collaboration and interdisciplinary teamwork among students.
- Apply design thinking methodologies to real-world engineering challenges.

Course Description

This course offers an exploration of the principles and methodologies behind design thinking, emphasizing on a human-centered approach to innovation and problem-solving. Through a blend of lectures, hands-on exercises, and interactive workshops, participants learn to empathize with users, define problems, generate creative solutions, and iterate on prototypes.

UNIT I Introduction to Design Thinking **2**

Definition and principles of design thinking – Importance and applications in engineering – Case studies of successful design thinking projects

UNIT II Empathize and Define **2**

Understanding user needs and motivations – Techniques for empathetic research (interviews, observations, etc.) – Defining problem statements based on user insights

UNIT III Ideate and Prototype **2**

Techniques for generating ideas (brainstorming, mind mapping, etc.) – Prototyping methods and tools – Iterative design process and feedback loops

UNIT IV Test and Iterate **2**

User testing and feedback collection – Analyzing and interpreting feedback – Iterating on prototypes based on feedback

UNIT V Application and Workshop **2**

Applying design thinking to engineering challenges – Workshop sessions for hands-on practice – Presentation (Posters / PPT / Demonstration) of final projects and reflection on the design process

Workshop Ideas:

1. Design Sprints: Conduct short, intensive workshops where students work collaboratively to solve a specific problem within a constrained timeframe.
2. Design Challenges: Pose open-ended design challenges to students and facilitate group work sessions where they brainstorm and prototype solutions.
3. User Persona Creation: Have students create user personas based on research findings and use them to guide the design process.
4. Prototyping Sessions: Provide materials and tools for students to create rapid prototypes of their ideas, encouraging experimentation and creativity.

5. Design Critiques: Organize sessions where students present their prototypes to peers for feedback and constructive criticism, fostering a culture of iteration and improvement.

Course Format

Lectures and discussions, Workshops, Group discussions and presentations,

COURSE OUTCOMES:

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

CO1: Understand the principles and process of design thinking.
CO2: Identify user needs through empathetic research.
CO3: Generate creative ideas and solutions through brainstorming and prototyping.
CO4: Apply design thinking methodologies to solve engineering problems effectively.
CO5: Work collaboratively in multidisciplinary teams to address complex challenges.

TEXT BOOKS:

1. Tim Brown, "Change by Design", Revised and Updated, Harper, 2019.
2. Christian Müller-Roterberg, “Handbook of Design Thinking: Tips & Tools for How to Design Thinking”, Independently Published, 2018.
3. Jeanne Liedtka, and Tim Ogilvie, “Designing for Growth: A Design Thinking Tool Kit for Managers”, Columbia University Press, 2011.

REFERENCES:

1. Don Norman, “The Design of Everyday Things”, Basic Books, 2015.
2. Hasso Plattner, Christoph Meinel, and Larry Leife (editors), “Design Thinking: Understand – Improve – Apply”, Springer-Verlag, 2013.
3. Richard Banfield, C. Todd Lombardo and Trace Wax, “Design Sprint: A Practical Guidebook for Building Great Digital Products”, O'Reilly Media, 2015.

TOTAL: 10 PERIODS

Mandatory Courses – II

Course Code	ENVIRONMENTAL SCIENCE AND SUSTAINABILITY	L	T	P	C
MC4401		2	0	0	0

COURSE OBJECTIVES:

- To introduce the basic concepts of environment, ecosystems and biodiversity and emphasize on the biodiversity of India and its conservation.
- To impart knowledge on the causes, effects and control or prevention measures of environmental pollution and natural disasters.
- To facilitate the understanding of global and Indian scenario of renewable and nonrenewable resources, causes of their degradation and measures to preserve them.
- To familiarize the concept of sustainable development goals and appreciate the interdependence of economic and social aspects of sustainability, recognize and analyze climate changes, concept of carbon credit and the challenges of environmental management.
- 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 **2**

Definition, scope and importance of environment – need for public awareness. Ecosystem 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 **2**

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 **2**

Energy management and conservation, New Energy Sources: Need of new sources. Different types new energy sources. Applications of- Hydrogen energy, Ocean energy resources, Tidal energy conversion. Concept, origin and power plants of geothermal energy.

UNIT IV SUSTAINABILITY AND MANAGEMENT **2**

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 **2**

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 : 10 PERIODS

Course Outcomes:

Upon Completion of the course the students will be able to

CO1: To recognize and understand the functions of the 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.

Text Books:

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.
5. Erach Bharucha "Textbook of Environmental Studies for Undergraduate Courses" Orient Blackswan Pvt. Ltd. 2013.

Course Code	WELL-BEING WITH TRADITIONAL PRACTICES-YOGA, AYURVEDA AND SIDDHA	L	T	P	C
MC4402		2	0	0	0

COURSE OBJECTIVES:

- To enjoy life happily with fun filled new style activities that help to maintain health also
- To adapt a few lifestyle changes that will prevent many health disorders
- To be cool and handbill every emotion very smoothly in every walk of life
- To learn to eat cost effective but healthy foods that are rich in essential nutrients
- To develop immunity naturally that will improve resistance against many health disorders

UNIT I HEALTH AND ITS IMPORTANCE 2

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 heath.
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 - Increase 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 BMI-Importance and actions to be taken

UNIT II DISASTER RISK REDUCTION (DRR) 2

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

UNIT III ROLE OF AYURVEDA & SIDDHA SYSTEMS IN MAINTAINING HEALTH 2

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 **2**

Emotional health - Definition and types - Three key elements: the subjective experience - the physiological response - the behavioral 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 2

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 yogasanas for cure and prevention of health disorders - What yoga can bring to our life.

TOTAL:10 PERIODS

Course Outcomes:

Upon Successful Completion of the course the students will be able to

CO1:Embrace enjoyable, health-promoting activities to foster happiness and well-being.

CO2: Adopt lifestyle changes aimed at preventing various health issues.

CO3: Cultivate emotional resilience to navigate life's challenges with ease.

CO4: Learn to incorporate cost-effective, nutrient-rich foods into daily dietary habits.

CO5: Enhance natural immunity to bolster resistance against a range of health disorders

Text Books:

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

Course Code	HISTORY OF SCIENCE AND TECHNOLOGY IN INDIA	L	T	P	C
MC4403		2	0	0	0

COURSE OBJECTIVES:

- Understand key historical concepts and perspectives relevant to the study of science and technology in India.
- Explore the historiography of Indian science and technology through the works of influential historians.
- Trace the development of science and technology in ancient and medieval India, including interactions with other civilizations.
- Analyze the impact of colonialism on Indian science and technology, including responses to Western influence.
- Examine the growth of techno-scientific institutions in colonial India

UNIT I CONCEPTS AND PERSPECTIVES 2

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, concept of historical inevitability, 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 2

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 2

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 2

Legacy of technology in Medieval India, Interactions with Arabs Development in medical knowledge, interaction between Unani and Ayurveda and alchemy Astronomy and Mathematics: interaction with Arabic Sciences Science and Technology on the eve of British conquest

UNIT V SCIENCE AND TECHNOLOGY IN COLONIAL INDIA 2

Science and the Empire Indian response to Western Science Growth of techno-scientific institutions

TOTAL:10 PERIODS

Course Outcomes:

Upon Successful Completion of the course the students will be able to

CO1: Understand historical concepts in the context of Indian science and technology.

CO2: Explore diverse historiography of Indian science and technology.

CO3: Trace the development of science and technology in ancient and medieval India

CO4: Analyze the impact of colonialism on Indian science and technology

CO5: Develop critical thinking skills to assess the relationship between science, technology, and society in India's history

Text Books:

1.A Social History of Indian Science" by Dhruv Raina

2. Science and Society in Early India" by D.P. Chattopadhyaya

References:

1. History of Science and Technology in India: Vol 1-5" by Debiprasad Chattopadhyaya
2. Science, Technology, Imperialism, and War" by Debiprasad Chattopadhyaya

Course Code	POLITICAL AND ECONOMIC THOUGHT FOR A HUMANE SOCIETY	L	T	P	C
MC4404		2	0	0	0

COURSE OBJECTIVES:

- This course will begin with a short overview of human needs and desires and how different political-economic systems try to fullfill 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 CAPITALISM 2

Capitalism – Free markets, demand-supply, perfect competition, laissez-faire, monopolies imperialism. Liberal democracy

UNIT II COMMUNISM 2

Fascism and totalitarianism. World war I and II. Cold war. Communism – Mode of production, theory of labour, surplus value, class struggle, dialectical materialism, historical materialism, Russian and Chinese models.

UNIT III WELFARE STATE 2

Welfare state. Relation with human desires. Empowered human beings, satisfaction. (3 lectures) Gandhian thought. Swaraj, Decentralized economy & polity, Community. Control over one's lives. Relationship with nature

UNIT IV ESSENTIAL ELEMENTS OF INDIAN CIVILIZATION 2

Technology as driver of society, Role of education in shaping of society. Future directions. (Refs: Nandkishore Acharya, David Dixon, Levis Mumford)

UNIT V SCIENCE AND TECHNOLOGY IN COLONIAL INDIA 2

Science and the Empire Indian response to Western Science Growth of techno-scientific institutions

TOTAL:10 PERIODS

Course Outcomes:

Upon Successful Completion of the course the students will be able to:

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.

Text Books:

1. Adam Smith, J.S.MILL

References:

1. A Nagaraj, M K Gandhi, JC Kumarappa

Course Code	STATE, NATION BUILDING AND POLITICS IN INDIA	L	T	P	C
MC4405		2	0	0	0

COURSE OBJECTIVES:

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.

UNIT I CONCEPTS AND PERSPECTIVES 2

Understanding the need and role of State and politics. Development of Nation-State, sovereignty, sovereignty in a globalized world.

UNIT II ORGANS OF STATE 2

Organs of State – Executive, Legislature, Judiciary. Separation of powers, forms of government unitary-federal, Presidential-Parliamentary, The idea of India.

UNIT III NATIONAL AWAKENING 2

1857 and the national awakening. 1885 Indian National Congress and development of national movement – its legacies.

UNIT IV CONSTITUTION 2

Constitution making and the Constitution of India. Goals, objective and philosophy. Need for a Federal system

UNIT V NATIONAL INTEGRATION AND NATION-BUILDING 2

Challenges of nation-building – State against democracy (Kothari) New social movements. The changing nature of Indian Political System, the future scenario.

TOTAL:10 PERIODS

Course Outcomes:

Upon Successful Completion of the course the students will be able to:

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.

Text Books:

1.Sunil Khilnani, The Idea of India. Penguin India Ltd., New Delhi.

2.Madhav Khosla, The Indian Constitution, Oxford University Press. New Delhi, 2012.

References:

1.Brij Kishore Sharma, Introduction to the Indian Constitution, PHI, New Delhi, latest edition.

2.Sumantra Bose, Transforming India: Challenges to the World's Largest Democracy, Picador India, 2013.

Course Code	INDUSTRIAL SAFETY	L	T	P	C
MC4406		2	0	0	0

COURSE OBJECTIVES:

- To Understand the Introduction and basic Terminologies safety.
- To enable the students to learn about the Important Statutory Regulations and standards.
- To enable students to Conduct and participate the various Safety activities in the Industry.
- To have knowledge about Workplace Exposures and Hazards.
- To assess the various Hazards and consequences through various Risk Assessment Techniques.

UNIT I SAFETY TERMINOLOGIES 2

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 2

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 2

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 2

Noise hazard- Particulate matter- musculoskeletal disorder improper sitting poster and lifting Ergonomics RULE & REBA- Unsafe act & Unsafe Condition- Electrical Hazards- Crane Safety- Toxic gas Release

UNIT V HAZARD IDENTIFICATION TECHNIQUES 2

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:10 PERIODS

Course Outcomes:

Upon Successful Completion of the 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:Analyze on the impact of Occupational Exposures and their Remedies

CO5:Obtain knowledge of Risk Assessment Techniques

Text Books:

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
5. Society of Safety Engineers, USA

Course Code	NCC CREDIT COURSE	L	T	P	C
AD42XX		2	0	0	2

UNIT I NCC General **6**

Aims, Objectives & Organization of NCC (1) – Incentives (2) – Duties of NCC Cadet (1) – NCC Camps: Types & Conduct (2)

UNIT II National Integration and Awareness **4**

National Integration: Importance & Necessity (1) – Factors Affecting National Integration (1) – Unity in Diversity & Role of NCC in Nation Building (1) –Threats to National Security (1)

UNIT III Personality Development **7**

Self-Awareness, Empathy, Critical & Creative Thinking, Decision Making and Problem Solving (2) – Communication Skills (3) – Group Discussion: Stress & Emotions (2)

UNIT IV Leadership **5**

Leadership Capsule: Traits, Indicators, Motivation, Moral Values, Honour Code (3) – Case Studies: Shivaji, Jhansi Ki Rani (2)

UNIT V Social Service and Community Development **8**

Basics, Rural Development Programmes, NGOs, Contribution of Youth (3) – Protection of Children and Women Safety (1) Road/ Rail Travel Safety (1) – New Initiatives (2) – Cyber and Mobile Security Awareness (1)

TOTAL: 30 PERIODS

Levels:

Level	**Activities**
1 ARMY	II semester: The NCC cadets are fresh to NCC activities are provided with above curriculum / syllabus
2 NAVY	For the Higher semester cadets of IV semester the credits are awarded with the basic qualifying needs Minimum one Combined Annual training Camp (CATC) or Attachment Camp or Centrally Organised Camp has to be attended (camp certificate is required) Appeared for B Certificate Exam
3 AIR FORCE	For the Higher semester cadets of VI semester the credits are awarded with the basic qualifying needs Minimum one Combined Annual training Camp (CATC) or Attachment Camp or Centrally Organised Camp has to be attended (camp certificate is required) Appeared for C Certificate Exam