B. TECH. PROGRAM CURRICULUM - 2025

(Applicable for the batches admitted from A.Y 2025-26)

Version 2.0

ARTIFICIAL INTELLIGENCE & MACHINE LEARNING

(in Collaboration with Microsoft)

Version 1.0–Approved in the BoS meeting on 17-05-2025 & Academic Council meeting on 14-06-2025

Version 2.0–Approved in the BoS meeting on 06-06-2026 & Academic Council meeting on 20-06-2026

ADITYA UNIVERSITY

Aditya Nagar, ADB Road, Surampalem - 533 437

ADITYA UNIVERSITY

B.Tech. in Department of Artificial Intelligence and Machine Learning

VISION

To be a leading hub in Artificial Intelligence through excellence in education, innovation, and sustainable practices.

MISSION

M1: Deliver cutting-edge AI education with advanced tools, techniques, and industry collaboration to address global challenges.

M2: Nurture creativity, research, and lifelong learning for personal and professional excellence.

M3: Promote ethical leadership and sustainability to build a responsible AI ecosystem.

PROGRAM EDUCATIONAL OBJECTIVES (PEOs)

PEO1: Apply emerging tools and technologies to solve interdisciplinary problems.

PEO2: Advance in education and research by applying innovative ideas and project-based learning.

PEO3: Work collaboratively and establish leadership with ethical, social, and financial accountability.

PROGRAM SPECIFIC OUTCOMES (PSOs)

PSO1: Apply core engineering, computing principles, AI and programming skills to develop efficient software solutions.

PSO2: Utilize emerging technologies to solve real-world challenges in industry and society with innovation and responsibility.

PROGRAM OUTCOMES (PO's)

The 11 Program Outcomes are described as below.

After successful completion of the program, the graduates will be able to

PO1 **Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO2 **Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences

PO3 **Design/development of solutions**: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations

PO4 **Conduct investigations of complex problems**: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions

PO5 **Engineering tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations

PO6 **The Engineer and the world :** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice

PO7 **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice

PO8 **Individual and collaborative teamwork**: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings

PO9 **Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO10 **Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO11 **Life-long learning**: Recognize the need for and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

ADITYA UNIVERSITY

Department of Artificial Intelligence and Machine Learning

B.Tech (AI&ML) in Collaboration with Microsoft
Program Curriculum-2025
(Applicable for the batches admitted from the A.Y.2025-26)

UG Programs Offered

- B. Tech in (Artificial Intelligence and Machine Learning)
- Minor Stream offered in B.Tech (Artificial Intelligence and Machine Learning) in Analytics and Security - Collaboration with Microsoft
- B. Tech in (Artificial Intelligence and Machine Learning) with
  - Minor degree in Civil Engineering.
  - Minor degree in Electrical and Electronics Engineering
  - Minor degree in Mechanical Engineering
  - Minor degree in Electronics and Communication Engineering
  - Minor degree in Petroleum Technology
  - Minor degree in Mining Engineering
  - Minor degree in Agricultural Engineering
  - Minor degree in Entrepreneurship Development and Incubation
  - Minor degree in Quantum Technologies

Credit Division Category-wise

S.No	Broad Category of Course	UGC	Credits
1	Major Core Courses (MCC)	80	85
2	Minor Stream Courses (MSC) (or) Minor Stream Courses (Min 20 credits) and University Open Elective Courses (UEC) (upto 12 credits)	32	32
3	Multidisciplinary Courses (MDC)	9	9
4	Ability Enhancement Courses (AEC)	8	9
5	Skill Enhancement Courses (SEC)	9	5
6	Value Added Courses (VAC)	6-8	4
7	Summer Internships (SI)	2-4	4
8	Full Semester Internship (PROJ)	12	12
9	Mandatory Courses (MC)		0
10	Certification Courses (CC)		0
Total Credits to be earned for B.Tech Degree		**160**	**160**

Foundation Courses – FC

Intermediate-level Courses - IC

Advanced Courses – AC

Major Core Courses (MCC)

Course Code	Course Name	Level	L	T	P	C	CIE	SEE	Total	Pre-requisite
2501MA01	Linear Algebra & Calculus	**FC**	2	1		3	50	50	100	-
2501MA02	Differential Equations & Vector Calculus	**FC**	2	1		3	50	50	100	-
2501CS71	Computer Organization & Architecture	**FC**	2	1		3	50	50	100	DLD
2501PH02	Modern Physics	**FC**	2		1	3	50	50	100	-
2501ME01	Engineering Graphics	**FC**	1		2	3	50	50	100	-
2501IT42	Programming with Python	**FC**	2		2	4	50	50	100	PPSC
2501CS01 / 2601DS01	Programming for Problem Solving Using C* / Problem Solving Through C Programming#	**FC**	2		2	4	50	50	100	-
2501CS03/ 2601DS02	Data Structures* / Data Structures using C++#	**FC**	2		2	4	50	50	100	PPSC
2501IT01/ 2601AI01	Business Intelligence Lab* / Business Intelligence & AI Tools Lab#	**FC**			2	2	50	50	100	-
2501MA08	Discrete Mathematics	IC	2	1		3	50	50	100	LAC
2501MA09	Probability & Statistics	IC	2	1		3	50	50	100	LAC
2501AI02	Artificial Intelligence	IC	2		1	3	50	50	100	-
2501CS08	Object Oriented Programming through C++*	IC	2		2	4	50	50	100	PPSC
2501IT05	Database Management Systems	IC	2		2	4	50	50	100	PPSC
2501AI03	Data Mining	IC	2		1	3	50	50	100	-
2501IT06	Java Programming	IC	2		2	4	50	50	100	PPSC
2601AI02	Full Stack Application Development using MERN#	IC	2		2	4	50	50	100	JP
2501CS09	Language Processors	IC	2	1		3	50	50	100	PPSC
2501IT07	Agile Software Engineering	IC	2		1	3	50	50	100	PPSC
2501CS13	Operating Systems	IC	2		1	3	50	50	100	-
2501CS07	Computer Networks	IC	2		1	3	50	50	100	-
2501AI05	Machine Learning	IC	2		2	4	50	50	100	DAE
2501CS10	Advanced Data Structures & Algorithm Analysis	AC	2		1	3	50	50	100	DS
2501AI11	Deep Learning	AC	2		2	4	50	50	100	DAE
2501AI20	Natural Language Processing & Prompt Engineering	AC	2		1	3	50	50	100	ML

2501AI21	Reinforcement Learning & Gen AI	AC	2		1	3	50	50	100	ML
2501AI04	Big Data Analytics	AC	2		1	3	50	50	100	DM
Total			**51**	**6**	**32**	**89**				

Multidisciplinary Courses (MDC)

Course Code	Course Name	Level	L	T	P	C	CIE	SEE	Total	Pre-requisite
2501EE01	Basic Electrical & Electronics Engineering	**FC**	2		2	4	50	50	100	-
2501EC95	Digital Logic Design	**FC**	2	1		3	50	50	100	-
2501MB01	Engineering Economics & Management	**FC**	2			2	50	50	100	-
Total			**6**	**1**	**2**	**9**				

Ability Enhancement Courses (AEC)

Course Code	Course Name	Level	L	T	P	C	CIE	SEE	Total	Pre-requisite
2501EN01	Essential Cognitive Skills for Engineers	**FC**			1	1	100	-	100	-
2501EN02	Advanced Cognitive Skills for Engineers	**FC**			1	1	100	-	100	ECSE-I
2501UC07	Design Thinking	**FC**			1	1	100	-	100	-
2501UC08	Universal Human Values	**FC**	2			2	100	-	100	-
2501UC09	Technical Paper Publication	AC			2	2	100	-	100	-
2501AI35	Student Activity Based Learning	AC				2				
Total			**2**		**5**	**9**				

Skill Enhancement Courses (SEC)

Course Code	Course Name	Level	L	T	P	C	CIE	SEE	Total	Pre-requisite
2501CS17	Object Oriented Analysis & Design using UML	IC			2	2	50	50	100	ASE
2501AI34/ 2601CS01	Web Application Development using MERN Stack*/ Linux Programming#	AC			2	2	50	50	100	-
2501CS19	CI/CD using DevOps	AC			1	1	100	-	100	-
Total					**5**	**5**				

Value Added Courses (VAC)

Course Code	Course Name	Level	L	T	P	C	CIE	SEE	Total	Pre-requisite
2501CS04	Internet of Things	**FC**			1	1	50	50	100	-
2501AI06	Data Analysis Essentials	IC			2	2	50	50	100	PP
2501UC11	Employability Skills-I	FC			3	0	100	-	100	-
2501UC13	Employability Skills-II	FC			3	0	100	-	100	ES-I
2501UC14	Employability Skills-III	IC			3	0	100	-	100	ES-II
2501UC15	Employability Skills-IV	IC			3	0	100	-	100	ES-III
2501UC16	Employability Skills-V	AC			3	1	100	-	100	ES-IV
Total					**18**	**4**				

Summer Internships (SI)

Course Code	Course Name	Level	L	T	P	C	CIE	SEE	Total	Pre-requisite
2501AI22	Summer Internship-I	IC			2	2	100	-	100	-
2501AI23	Summer Internship-II	AC			2	2	100	-	100	-
Total					**4**	**4**				

Full Semester Internship (PROJ)

Course Code	Course Name	Level	L	T	P	C	CIE	SEE	Total	Pre-requisite
2501AI25	Full Semester Internship	AC			12	12	50	50	100	-
Total					**12**	**12**				

Mandatory Courses (MC)

Course Code	Course Name	Level	L	T	P	C	CIE	SEE	Total	Pre-requisite
2501AC01	Environmental Science	**FC**	2			0	100	-	100	-
2501AC02	Constitution of India	**FC**	2			0	100	-	100	-
2501AC03	Research Methodology	**FC**	2			0	100	-	100	-
2501AC04	Intellectual Property Rights & Patents	**FC**	2			0	100	-	100	-
2501AC05	Indian Knowledge Systems	**FC**	2			0	100	-	100	-
Total			**10**			**0**				

Minor Stream: Analytics and Security (In Collaboration with Microsoft)

Course Code	Course Name	Level	L	T	P	C	CIE	SEE	Total	Pre-requisite
2501AI36	Network Security Essentials	FC	2			2	50	50	100	-
2501AI37	Introduction to System Design	IC	2		1	3	50	50	100	-
2501AI38	Introduction to Generative AI	IC	2		1	3	50	50	100	-
2501AI39	Cloud Security Essentials	IC	2		1	3	50	50	100	-
2501AI40	Cyber Security Essentials	IC	2		1	3	50	50	100	-
2501AI41	Fundamentals of Block Chain	**IC**	2		1	3	50	50	100	-
2501AI42	AI for Cyber Threat Intelligence	IC	2		1	3	50	50	100	-
2501AI43	Generative Adversarial Networks Fundamentals	AC	2		1	3	50	50	100	-
2501AI44	AI in Healthcare Security	AC	2		1	3	50	50	100	-
2501AI45	Advanced Topics in Generative AI	AC	2		1	3	50	50	100	-
2501AI46	Ethical AI and Responsible Computing	AC	2		1	3	50	50	100	-
Total					**32**	**32**				

Certification Courses

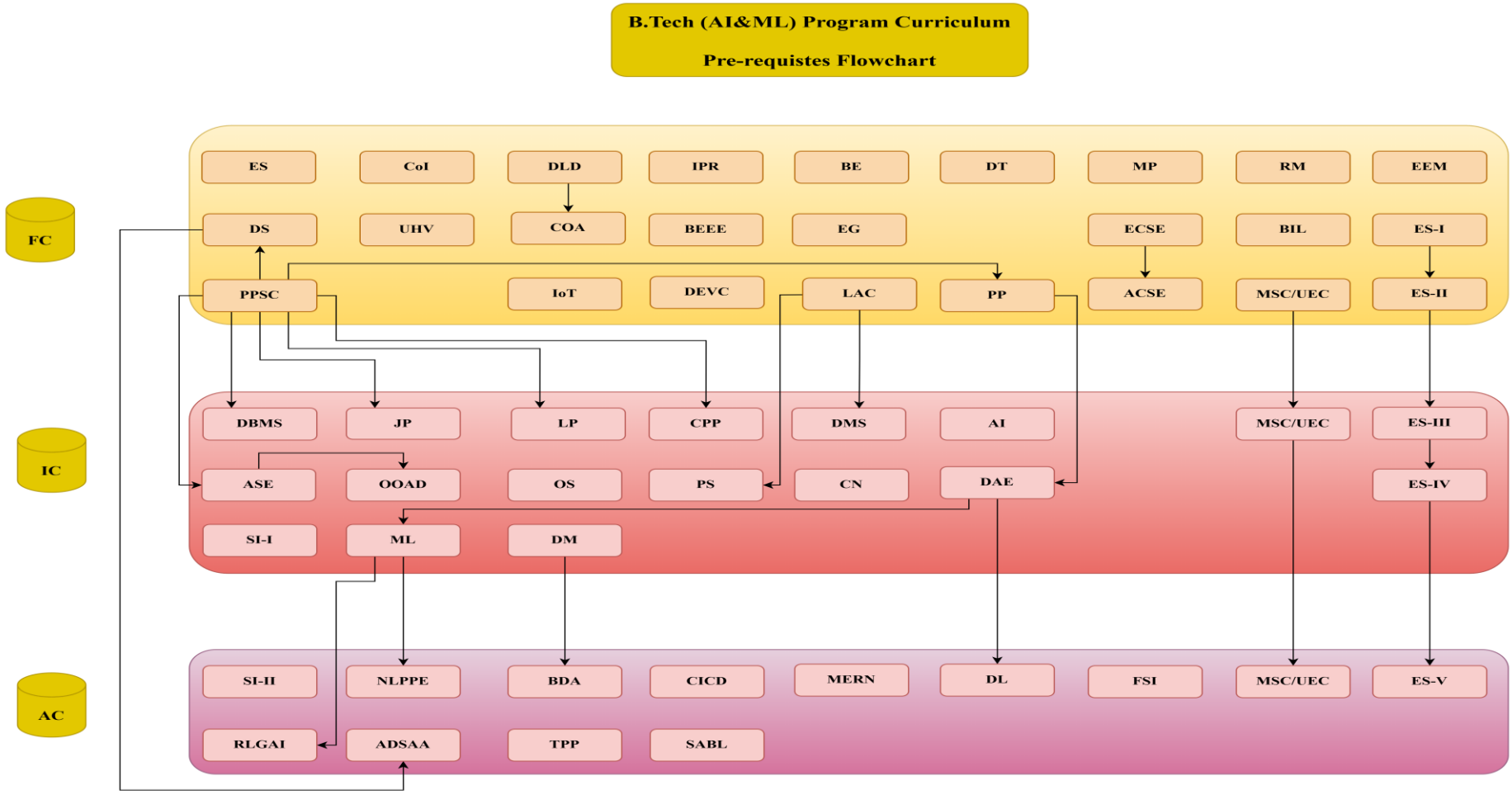
Course Code	Course Name	Level	L	T	P	C	CIE	SEE	Total	Pre-requisite
	Project Based Learning – Certification AZ-900 : Microsoft Azure Fundamentals	AC			2					-
	Project Based Learning - Certification AI-901: Azure AI Fundamentals	AC			2					-
	Project Based Learning - Certification DP-900: Azure Data Fundamentals	AC			2					-
	Project Based Learning - Certification AI-103: Azure AI App and Agent Developer	AC			2					-
	Project Based Learning - Certification AI-300: Azure MLOps Engineer	AC			2					-
	Project Based Learning - Certification AI-200: Azure AI Cloud Developer	AC			2					-
	Project Based Learning - Certification DP-800: Azure SQL AI Developer Associate	AC			2					-
Total					**14**					

Foundation Courses (FC)					
Course Title	**Category**	**L**	**T**	**P**	**C**
Linear Algebra & Calculus	MCC	2	1		3
Differential Equations & Vector Calculus	MCC	2	1		3
Computer Organization & Architecture	MCC	2	1		3
Modern Physics	MCC	2		1	3
Engineering Graphics	MCC	1		2	3
Programming with Python	MCC	2		2	4
Programming for Problem Solving Using C	MCC	2		2	4
Data Structures	MCC	2		2	4
Business Intelligence Lab	MCC			2	2
Basic Electrical & Electronics Engineering	MDC	2		2	4
Digital Logic Design	MDC	2	1		3
Engineering Economics & Management	MDC	2			2
Essential Cognitive Skills for Engineers	AEC			1	1
Advanced Cognitive Skills for Engineers	AEC			1	1
Design Thinking	AEC			1	1
Universal Human Values	AEC	2			2
Internet of Things	VAC			1	1
Employability Skills-I	VAC			3	0
Employability Skills-II	VAC			3	0
Environmental Science	MC	2			0
Constitution of India	MC	2			0
Research Methodology	MC	2			0
Intellectual Property Rights & Patents	MC	2			0
Biology for Engineers	MC	2			0
Courses from Minor Stream(s)	MSC				3

Intermediate-Level Courses (IC)					
Course Title	**Category**	**L**	**T**	**P**	**C**
Discrete Mathematics	MCC	2	1		3
Probability & Statistics	MCC	2	1		3
Artificial Intelligence	MCC	2		1	3
Object Oriented Proramming through C++	MCC	2		2	4
Database Management Systems	MCC	2		2	4
Data Mining	MCC	1		2	3
Java Programming	MCC	2		2	4
Language Processors	MCC	2	1		3
Agile Software Engineering	MCC	2		1	3
Operating Systems	MCC	2		1	3
Computer Networks	MCC	2		1	3
Machine Learning	MCC	2		2	4
Object Oriented Analysis & Design using UML	SEC			2	2
Data Analysis Essentials	VAC			2	2
Employability Skills-III	VAC			3	0
Employability Skills-IV	VAC			3	0
Summer Internship – I	SI			2	2
Courses from Minor Stream(s)	MSC				3

Advanced Courses (AC)					
Course Title	**Category**	**L**	**T**	**P**	**C**
Advanced Data Structures & Algorithm Analysis	MCC	2		1	3
Deep Learning	MCC	2		2	4
Natural Language Processing & Prompt Engineering	MCC	1		2	3
Reinforcement Learning & Gen AI	MCC	1		2	3
Big Data Analytics	MCC	2		1	3
Technical Paper Publication	AEC			2	2
Student Activity Based Learning	AEC				2
Web Application Development using MERN Stack	SEC			2	2
CI/CD using DevOps	SEC			1	1
Employability Skills-V	VAC			3	1
Summer Internship – II	SI			2	2
Full Semester Internship	PROJ			12	12
Courses from Minor Stream(s)	MSC				3

Note: *Courses are applicable only to students admitted in the year 2025.
#Courses are applicable to the students admitted from the year 2026 onwards.
All the other courses are applicable to students admitted from the year 2025 onwards


B.Tech (AI&ML) Program Curriculum
Pre-requistes Flowchart
FC
IC
AC
ES
CoI
DLD
IPR
BE
DT
MP
RM
EEM
DS
UHV
COA
BEEE
EG
ECSE
BIL
ES-I
PPSC
IoT
DEVC
LAC
PP
ACSE
MSC/UEC
ES-II
DBMS
JP
LP
CPP
DMS
AI
MSC/UEC
ES-III
ASE
OOAD
OS
PS
CN
DAE
ES-IV
SI-I
ML
DM
SI-II
NLPPE
BDA
CICD
MERN
DL
FSI
MSC/UEC
ES-V
RLGAI
ADSAA
TPP
SABL

FC-Foundation Courses

ES-Environmental Science
CoI-Constitution of India
DLD-Digital Logic Design
IPR- Intellectual Property Rights & Patents
BE- Biology for Engineers
DT- Design Thinking
MP- Modern Physics
RM-Research Methodology
EEM- Engineering Economics & Management
DS- Data Structures
UHV- Universal Human Values
COA- Computer Organization & Architecture
BEEE- Basic Electrical & Electronics Engineering
EG- Engineering Graphics
ECSE- Essential Cognitive Skills for Engineers
ITAI- Business Intelligence Lab
ES-I- Employability Skills-I
PPSC- Programming for Problem Solving Using C
IoT- Internet of Things
DEVC- Differential Equations & Vector Calculus
LAC- Linear Algebra & Calculus
PP- Programming with Python
ES-II-Employability Skills-II

IC-Intermediate-Level Courses

DBMS- Database Management Systems
JP- Java Programming
LP- Language Processors
CPP-Object Oriented programming through C++
DMS- Discrete Mathematics
AI- Artificial Intelligence
ES-III- Employability Skills-III
ASE- Agile Software Engineering
OOAD- Object Oriented Analysis & Design using UML
OS- Operating Systems
PS- Probability & Statistics
CN- Computer Networks
DAE- Data Analysis Essentials
ACSE-Advanced Cognitive Skills for Engineers
ES-IV-Employability Skills-IV
SI-1- Summer Internship – I
ML- Machine Learning
DM- Data Mining

AC-Advanced Courses

SI-II- Summer Internship – II
NLPPE- Natural Language Processing & Prompt Engineering
BDA- Big Data Analytics
CI/CD- CI/CD using DevOps
MERN- Web Application Development using MERN Stack
DL- Deep Learning
FSI- Full Semester Internship
ES-V-Employability Skills-V
RLGAI- Reinforcement Learning & Gen AI
ADSAA- Advanced Data Structures & Algorithm Analysis
TPP- Technical Paper Publication
SABL-Student Activity Based Learning

MSC- Minor Stream Courses

Suggestive Semester-wise Curriculum

I SEMESTER

Course code	Course Title	Course		Credits				Total Hours
		Category	Level	L	T	P	Total	
2501MA01	Linear Algebra & Calculus	MCC	FC	2	1		3	3
2501CS01 / 2601DS01	Programming for Problem Solving Using C* / Problem Solving Through C Programming#	MCC	FC	2		2	4	6
2501IT01/ 2601AI01	Business Intelligence Lab* / Business Intelligence & AI Tools Lab#	MCC	FC			2	2	4
2501EC95	Digital Logic Design	MDC	FC	2	1		3	4
2501EE01	Basic Electrical & Electronics Engineering	MDC	FC	2		2	4	6
2501EN01	Essential Cognitive Skills for Engineers	AEC	FC			1	1	2
2501UC08	Universal Human Values	AEC	FC	2			2	2
	Project Based Learning – Certification AZ-900 : Microsoft Azure Fundamentals	CC	FC			2	0	4
Total				**10**	**2**	**9**	**19**	**31**

II SEMESTER

Course code	Course Title	Course		Credits				Total Hours
		Category	Level	L	T	P	Total	
2501MA02	Differential Equations & Vector Calculus	MCC	FC	2	1		3	3
2501PH02	Modern Physics	MCC	FC	2		1	3	4
2501CS03/ 2601DS02	Data Structures* / Data Structures using C++#	MCC	FC	2		2	4	6
2501ME01	Engineering Graphics	MCC	FC	1		2	3	5
2501CS71	Computer Organization & Architecture	MCC	FC	2	1		3	3
2501IT42	Programming with Python	MCC	IC	2		2	4	6
2501EN02	Advanced Cognitive Skills for Engineers	AEC	FC			1	1	2
2501UC07	Design Thinking	AEC	FC			1	1	2
2501UC11	Employability Skills-I	VAC	FC			3	0	*6*
2501AC01	Environmental Science	MC	FC	2			0	2
	Project Based Learning - Certification AI-901: Azure AI Fundamentals	CC	FC			2	0	4
Total				**13**	**2**	**14**	**22**	**43**

III SEMESTER

Course code	Course Title	Course		Credits				Total Hours
		Category	Level	L	T	P	Total	
2501MA09	Probability & Statistics	MCC	IC	2	1		3	3
2501CS10	Advanced Data Structures & Algorithm Analysis	MCC	AC	2		1	3	4
2501AI02	Artificial Intelligence	MCC	IC	2		1	3	4
2501CS08/2501IT06	Object Oriented Programming through C++*/Java Programming#	MCC	IC	2		2	4	6
2501IT05	Database Management Systems	MCC	IC	2		2	4	6
2501IT07	Agile Software Engineering	MCC	IC	2		1	3	4
2501AI06	Data Analysis Essentials	VAC	IC			2	2	4
2501UC13	Employability Skills-II	VAC	FC			3	0	*6*
2501AC02	Constitution of India	MC	FC	2			0	*2*
	Project Based Learning - Certification DP-900 : Azure Data Fundamentals	CC	IC			2	0	4
Total				**14**	**1**	**14**	**22**	**43**

IV SEMESTER

Course Code	Course Title	Course		Credits				Total Hours
		Category	Level	L	T	P	Total	
2501MA08	Discrete Mathematics	MCC	IC	2	1		3	3
2501AI03	Data Mining	MCC	IC	2		1	3	4
2501IT06/ 2601AI02	Java Programming*/ Full Stack Application Development using MERN#	MCC	IC	2		2	4	6
2501CS13	Operating Systems	MCC	IC	2		1	3	4
2501AI36	Minor Stream Course-1 Network Security Essentials	MSC	FC	2			2	2
2501AI37	Minor Stream Course-2 Introduction to System Design	MSC	FC	2		1	3	4
2501AI34/ 2601CS01	Web Application Development using MERN Stack*/ Linux Programming#	SEC	AC			2	2	4
2501UC14	Employability Skills-III	VAC	FC			3	0	6
2501AC03	Research Methodology	MC	FC	2			0	*2*
	Project Based Learning - Certification AI-103 : Azure AI App and Agent Developer	CC	IC			2	0	4
Total				**14**	**1**	**12**	**20**	**39**

V SEMESTER

Course code	Course Title	Course		Credits				Total Hours
		Category	Level	L	T	P	Total	
2501CS07	Computer Networks	MCC	IC	2		1	3	4
2501AI05	Machine Learning	MCC	IC	2		2	4	6
2501AI38	Minor Stream Course-3 Introduction to Generative AI	MSC	IC	2		1	3	4
2501AI39	Minor Stream Course-4 Cloud Security Essentials	MSC	IC	2		1	3	4
2501AI40	Minor Stream Course-5 Cyber Security Essentials	MSC	IC	2		1	3	4
2501MB01	Engineering Economics & Management	MDC	FC	2			2	2
2501CS17	Object Oriented Analysis & Design using UML	SEC	IC			2	2	4
2501UC15	Employability Skills-IV	VAC	FC			3	0	6
2501AI22	Summer Internship-I	SI	IC			2	2	4
2501AC04	Intellectual Property Rights & Patents	MC	FC	2			0	*2*
	Project Based Learning - Certification AI-300 : Azure MLOps Engineer	CC	IC			2	0	4
Total				**14**	**0**	**15**	**22**	**44**

VI SEMESTER

Course code	Course Title	Course		Credits				Total Hours
		Category	Level	L	T	P	Total	
2501AI11	Deep Learning	MCC	AC	2		2	4	6
2501AI20	Natural Language Processing & Prompt Engineering	MCC	AC	2		1	3	4
2501CS09	Language Processors	MCC	IC	2	1		3	3
2501AI41	Minor Stream Course-6 Fundamentals of Block Chain	MSC	IC	2		1	3	4
2501AI42	Minor Stream Course-7 AI for Cyber Threat Intelligence	MSC	IC	2		1	3	4
2501AI43	Minor Stream Course-8 Generative Adversarial Networks Fundamentals	MSC	IC	2		1	3	4
2501CS19	CI/CD using DevOps	SEC	AC			1	1	2
2501CS04	Internet of Things	VAC	FC			1	1	2
2501AC05	Indian Knowledge Systems	MC	FC	2			0	*2*
	Project Based Learning - Certification AI-200 : Azure AI Cloud Developer	CC	AC			2	0	4
Total				**14**	**1**	**10**	**21**	**35**

VII SEMESTER

Course code	Course Title	Course		Credits				Total Hours
		Category	Level	L	T	P	Total	
2501AI21	Reinforcement Learning & Gen AI	MCC	AC	2		1	3	4
2501AI04	Big Data Analytics	MCC	AC	2		1	3	4
2501AI44	Minor Stream Course-9 AI in Healthcare Security	MSC	AC	2		1	3	4
2501AI45	Minor Stream Course-10 Advanced Topics in Generative AI	MSC	AC	2		1	3	4
2501AI46	Minor Stream Course-11 Ethical AI and Responsible Computing	MSC	AC	2		1	3	4
2501AI23	Summer Internship-II	SI	AC			2	2	4
	Project Based Learning - Certification DP-800: Azure SQL AI Developer Associate	CC	AC			2	0	4
Total				**10**	**0**	**9**	**17**	**28**

VIII SEMESTER

Course code	Course Title	Course		Credits				Total Hours
		Category	Level	L	T	P	Total	
2501UC09	Technical Paper Publication	AEC	AC			2	2	4
2501UC16	Employability Skills-V	VAC	FC			3	1	**6**
2501AI35	Student Activity Based Learning	AEC	AC				2	4
2501AI25	Full Semester Internship	PROJ	AC			12	12	24
Total				**0**	**0**	**17**	**17**	**38**

Total Credit: 160.

Note: *Courses are applicable only to students admitted in the year 2025.
#Courses are applicable to the students admitted from the year 2026 onwards.
All the other courses are applicable to students admitted from the year 2025 onwards

*** To acquire a minor degree, a student has to earn 20 credits in addition to the 160 credits.**

Minor Degree in Civil Engineering

Course Code	Course Name	Level	L	T	P	C	CIE	SEE	Total	Pre-requisite
2501CE25	Repair & Rehabilitation of Structures	FC	3			3	50	50	100	-
2501CE43	Building Planning & Computer-Aided Drawing	FC			2	2	50	50	100	-
2501CE27	Green Buildings	FC	3			3	50	50	100	-
2501CE40	Fundamentals of Soil Behaviour	FC	2	1		3	50	50	100	-
2501CE54 2501CE47	Railway Engineering (or) Docks & Harbour Engineering	FC	3			3	50	50	100	-
2501CE36 2501CE37	Environmental Impact & Risk Management (or) Environmental Management	IC	3			3	50	50	100	-
2501CE56 2501CE49	Urban Transportation Planning (or) Intelligent Transportation Systems.	IC	3			3	50	50	100	-
TOTAL			**17**	**1**	**2**	**20**				

Minor Degree in Electrical and Electronics Engineering

Course Code	Course Name	Level	L	T	P	C	CIE	SEE	Total	Pre-requisite
2501EE55	Operation & Control of Electric machines	FC	2			2	50	50	100	BEEE
2501EE56	Fundamentals of Power Electronics	FC	2			2	50	50	100	BEEE
2501EE13	Electrical Measurements & Instrumentation	FC	2	1	1	4	50	50	100	ENA-1/BEEE
2501EE53	Electric Power Generation, Transmission & Distribution Systems	IC	3			3	50	50	100	ENA-1/BEEE
2501EE34 2501EE27	Alternative Energy Sources (or) Utilization of Electrical Energy	IC	3			3	50	50	100	EPGDS/ BEEE/ ISM
2501EE37 2501EE35	Hybrid Electric Vehicles (or) Special Electric machines	AC	3			3	50	50	100	FPE/ OCEM
2501EE43 2501EE30	Electrical Safety (or) Methods & Algorithms for Intelligent Control	AC	3			3	50	50	100	EPGDS/ PSA
TOTAL			**18**	**1**	**1**	**20**				

Minor Degree in Mechanical Engineering

Course Code	Course Name	Level	L	T	P	C	CIE	SEE	Total	Pre-requisite
2501ME74	Basic Mechanical Engineering	FC	2			2	50	50	100	-
2501ME04	Engineering Thermodynamics	IC	2	1		3	50	50	100	SSP/ MP
2501ME77	Introduction to Automobile Engineering (or) Mechanics of Solids	IC	3			3	50	50	100	SSP/ MP
2501ME78										
2501ME12	Heat Power Engineering (or) Refrigeration & Air	IC	2	1		3	50	50	100	ETD
2501ME40										
2501ME75	Production Technology	IC	3			3	50	50	100	EW
2501ME76	Metallurgy & Material Science	IC	3			3	50	50	100	SSP/ MP
2501ME79	Theory of Machines (or) Advanced Engineering Metrology	AC	3			3	50	50	100	SSP/ MP
2501ME80										
	TOTAL		**18**	**2**		**20**				

Minor Degree in Electronics and Communication Engineering

Course Code	Course Name	Level	L	T	P	C	CIE	SEE	Total	Pre-requisite
2501EC87	Fundamentals of Communications	FC	2		1	3	50	50	100	-
2501EC88	Fundamentals of Signal Processing	FC	2		1	3	50	50	100	-
2501EC89	Analog & Digital Circuits	IC	2			2	50	50	100	BEEE
2501EC42	Wireless LAN's & PAN's	IC	2	1	0	3	50	50	100	-
2501EC90	Linear & Digital IC Applications (or) Sensors & Actuators	IC	3			3	50	50	100	ADC
2501EC91										
2501EC92	Embedded Microcontrollers (or) Digital System Design	IC	2		1	3	50	50	100	ADC, PPSC
2501EC93										
2501EC67	Introduction to Internet of things (or) Modern Wireless Communications	AC	2		1	3	50	50	100	EM, WLAN's& PAN's
2501EC74			3			3	50	50	100	FC
TOTAL			15	1	4	20				

Minor Degree in Petroleum Technology

Course Code	Course Name	Level	L	T	P	C	CIE	SEE	Total	Pre-requisite
2501PT27	Introduction to Petroleum Engineering	FC	3			3	50	50	100	-
2501PT35	Unit Operations in Petroleum Industry	FC	3			3	50	50	100	-
2501PT47	Fundamentals of Geology & Reservoir Engineering	IC	3			3	50	50	100	-
2501PT48	Fundamentals of Drilling & Production Engineering (or)	IC	2			2	50	50	100	-
2501PT16	Unconventional Hydrocarbon Resources		2			2				
2501PT49	Natural Gas Hydrates (or)	AC	2			2	50	50	100	-
2501PT05	Fundamentals of Liquefied Natural Gas		2			2				
2501PT50	Artificial Lift Techniques (or)	AC	3			3	50	50	100	-
2501PT03	Enhanced Oil Recovery		2	1						
2501PT12	Petroleum Refinery Engineering	AC	2		2	4	50	50	100	-
TOTAL			**17**	**1**	**2**	**20**				

Minor Degree in Mining Engineering

Course Code	Course Name	Level	L	T	P	C	CIE	SEE	Total	Pre-requisite
2501MN03	Development of Mineral Deposits	FC	3			3	50	50	100	-
2501MN41	Green Mining	FC	3			3	50	50	100	-
2501MN06	Surface Mining	IC	3			3	50	50	100	DMD
2501MN24	Drilling & Blasting	IC	3			3			100	DMD
2501MN07	Underground Coal Mining Technology (or) Underground Metal Mining Technology	IC	3			3	50	50	100	DMD
2501MN08										
2501MN14	Mine Legislation & General Safety (or) Environmental Pollution & Control	AC	3			3	50	50	100	UCMT / UMMT
2501MN28										
2501MN45	Industrial Safety Practices (or) Ground Control	AC	2			2	50	50	100	-
2501MN46										
TOTAL			**20**			**20**				

Minor Degree in Agricultural Engineering

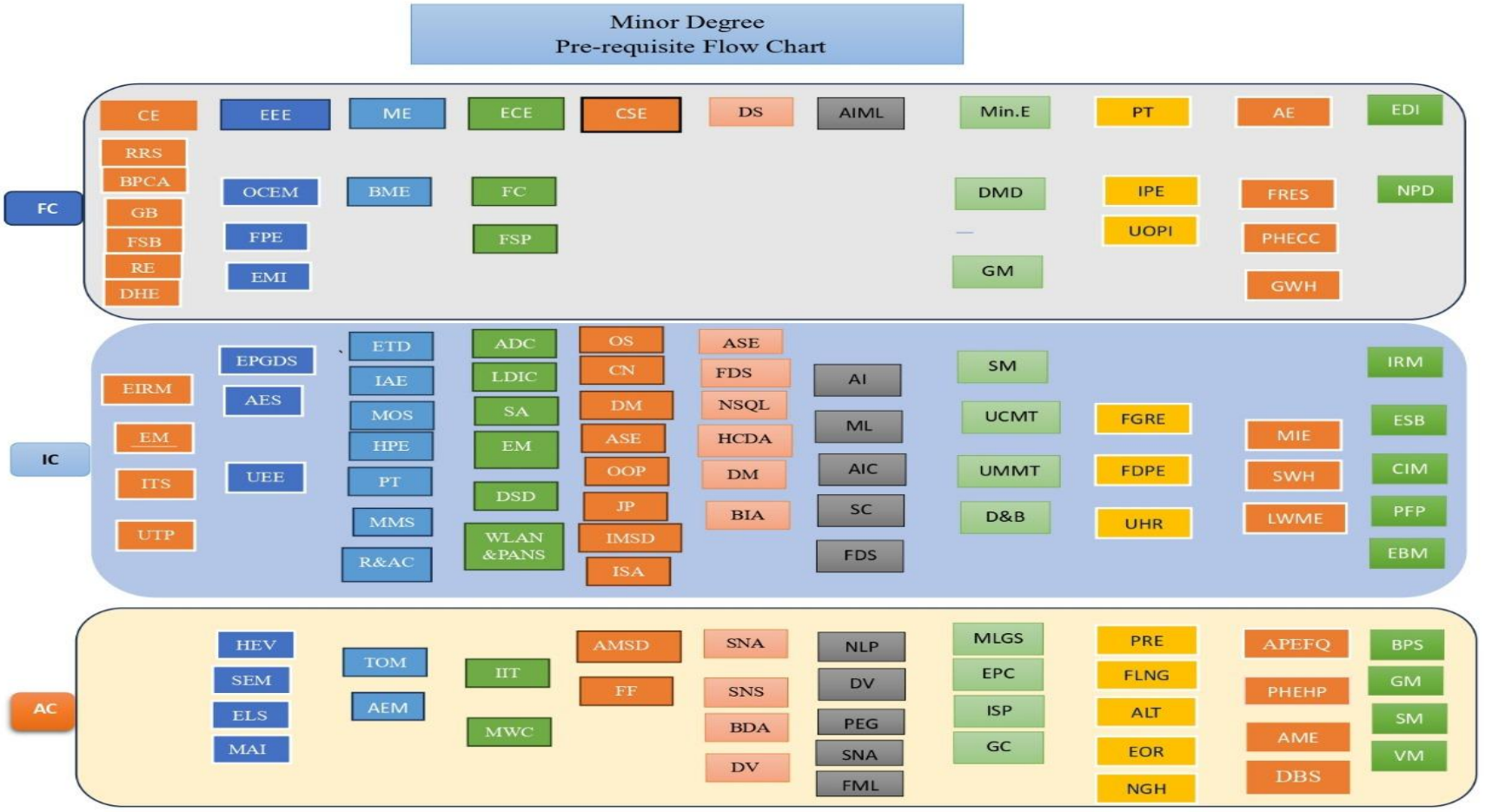
Course Code	Course Name	Level	L	T	P	C	CIE	SEE	Total	Pre-requisite
2501AE82	Fundamentals of Renewable Energy Sources	FC	3			3	50	50	100	
2501AE83	Post-harvest Engineering of Cereal Crops	FC	3			3	50	50	100	
2501AE84	Ground Water Hydrology	FC	3			3	50	50	100	
2501AE85	Micro Irrigation Systems	IC	2			2	50	50	100	
2501AE86	Surface Water Hydrology (or) Land & Water Management Engineering	IC	3			3	50	50	100	GWH
2501AE87										
2501AE88	Agricultural Process Engineering & Food Quality (or) Post-harvest Engineering for Horticultural Produce	AC	3			3	50	50	100	PHECC
2501AE89										
2501AE90	Agricultural Machinery & Equipment (or) Design of Bio-energy systems	AC	3			3	50	50	100	FRES
2501AE91										
TOTAL			**20**			**20**				

Minor Degree in Entrepreneurship Development & Incubation

Course Code	Course Name	Level	L	T	P	C	CIE	SEE	Total	Pre-requisite
2501MB07	New Product Development	FC	3			3	50	50	100	-
2501MB08	Entrepreneurship & Small Business Management	IC	2			2	50	50	100	-
2501MB09	Insurance & Risk Management	IC	3			3	50	50	100	-
2501MB10	Change & Innovations Management	IC	3			3	50	50	100	-
2501MB11	Personal Financial Planning (or) E-Business management	IC	3			3	50	50	100	-
2501MB12										
2501MB13	Business Policy & Strategic Management (or) Green Marketing	AC	3			3	50	50	100	-
2501MB14										
2501MB15	Startup Management (or) Venture Management	AC	3			3	50	50	100	-
2501MB16										
TOTAL			**20**			**20**				

Minor Degree in Quantum Technologies

S.No.	Course Code	Course Name	L	T	P	C	Semester
Mandatory Courses							
1	251EC097	Survey of Quantum technologies and Application	3	0	0	3	IV
2	251EC098	Foundations of Quantum Technologies	3	0	0	3	V
3	251EC099	Basic Programming Lab (or)	1	0	2	3	V
	251EC100	Basic Laboratory Course for Quantum Technologies					
4	251EC101	Quantum Algorithms and Cryptography	12 week 3 Credit - NPTEL MOOC			3	VII/VIII
Any One course from the below							
5	251EC102	Introduction to Quantum Computation	3	0	0	3	VI
6	251EC103	Introduction to Quantum Communication	3	0	0	3	VI
7	251EC104	Introduction to Quantum Sensing	3	0	0	3	VI
8	251EC105	Introduction to Quantum Materials	3	0	0	3	VI
Any One course from the below							
9	251EC106	Engineering Foundations of Quantum Technologies	3	0	0	3	VII
10	251EC107	Solid State Physics for Quantum Technologies	3	0	0	3	VII
11	251EC108	Quantum Optics	3	0	0	3	VII
12	251EC109	Quantum Cybersecurity	3	0	0	3	VII
13	251EC110	Quantum Machine Learning	3	0	0	3	VII
Total			**18**	**0**	**0**	**18**	


Minor Degree
Pre-requisite Flow Chart
FC
CE
RRS
BPCA
GB
FSB
RE
DHE
EEE
OCEM
FPE
EMI
ME
BME
ECE
FC
FSP
CSE
DS
AIML
Min.E
DMD
GM
PT
IPE
UOPI
AE
FRES
PHECC
GWH
EDI
NPD
IC
EIRM
EM
ITS
UTP
EPGDS
AES
UEE
ETD
IAE
MOS
HPE
PT
MMS
R&AC
ADC
LDIC
SA
EM
DSD
WLAN &PANS
OS
CN
DM
ASE
OOP
JP
IMSD
ISA
ASE
FDS
NSQL
HCDA
DM
BIA
AI
ML
AIC
SC
FDS
SM
UCMT
UMMT
D&B
FGRE
FDPE
UHR
MIE
SWH
LWME
IRM
ESB
CIM
PFP
EBM
AC
HEV
SEM
ELS
MAI
TOM
AEM
IIT
MWC
AMSD
FF
SNA
SNS
BDA
DV
NLP
DV
PEG
SNA
FML
MLGS
EPC
ISP
GC
PRE
FLNG
ALT
EOR
NGH
APEFQ
PHEHP
AME
DBS
BPS
GM
SM
VM

Dept.	FOUNDATION COURSE		INTERMEDIATE- LEVEL COURSE		ADVANCED COURSE	
CE	RRS	Repair & Rehabilitation of Structures	EIRM	Environmental Impact & Risk Management		
	BPCA	Building Planning & Computer-Aided Drawing	EM	Environmental Management		
	GB	Green Building	UTP	Urban Transportation Planning		
	FSB	Fundamental of Soil Behaviour	ITS	Intelligent Transportation Systems		
	RE	Railway Engineering				
	DHE	Docks & Harbour Engineering				
EEE	OCEM	Operation control of Electric Machines	AES	Alternative Energy Sources	MAI	Methods &Algorithms for Intelligent Control
	FPE	Fundamentals of Power Electronics	EPGDS	Electric Power Generation & Distribution Systems	HEV	Hybrid Electric Vehicles
	EMI	Electrical Measurements & Instrumentation	UEE	Utilization of Electrical Energy	SEM	Special Electric Machines
					ES	Electrical Safety
ME	BME	Basic Mechanical Engineering	ETD	Engineering Thermodynamics	TOM	Theory of Machines
			IAE	Introduction to Automobile Engineering	AEM	Advanced Engineering Metrology
			MOS	Mechanics of Solids		
			HPE	Heat Power Engineering		
			PT	Production Technology		
			MMS	Metallurgy & Material Science		
			R&AC	Refrigeration & Air Conditioning		
ECE	FC	Fundamentals of Communications	ADC	Analog & Digital Circuits	IIT	Introduction to Internet of things
			LDIC	Linear & Digital IC Applications		
			SA	Sensors & Actuators		
	FSP	Fundamentals of Signal Processing	EM	Embedded Microcontrollers	MWC	Modern Wireless Communications
			DSD	Digital System Design		
			WLAN &PAN	Wireless LANS & PANS		
CSE			ASE	Agile Software Engineering	WADMS	Web Application Development using MERN Stack
			OOP	Object Oriented Programming through C++		
			OS	Operating Systems	FF	Flutter Fundamentals
			CN	Computer Networks		
			JP	Java Programming		
			PC++	Programming with C++		
			ISA	Information Security Analysis and Audit		
			DM	Data Mining		
DS			DM	Data Mining	SNA	Social Network Analysis
			FDS	Fundamentals of Data Science	DV	Data Visualization
			ASE	Agile Software Engineering	BDA	Big Data Analytics
			NSQL	NoSQL Databases	SNSW	Social Networks and Semantic Web

			BIA	Business Intelligence & Analytics		
			HCDA	Health Care Data Analysis		
AIML			ML	Machine Learning	DV	Data Visualization
			AI	Artificial Intelligence	PEG	Prompt Engineering and GenAI
			AI C	AI Chatbot	SNA	Social Network Analysis
			FDS	Fundamentals of Data Science	FML	Federated Machine Learning
			SC	Soft Computing	NLP	Natural Language Processing
Min.E	DMD	Development of Mineral Deposits	SM	Surface Mining	MLGS	Mine Legislation and General Safety
	GM	Green Mining	UCMT	Underground Coal Mining Technology	EPC	Environmental Pollution & Control
			UMMT	Underground Metal Mining Technology	ISP	Industrial Safety Practices
			DB	Drilling & Blasting	GC	Ground Control
PT	IPE	Introduction to Petroleum Engineering	FGRE	Fundamentals of Geology and Reservoir Engineering	PRE	Petroleum Refinery Engineering
	UOPI	Unit operations in Petroleum Industry	FDPE	Fundamentals of Drilling and Production Engineering	FLNG	Fundamentals of Liquefied Natural Gas
					NGH	Natural Gas Hydrates
			UHR	Unconventional Hydrocarbon Resources	ALT	Artificial Lift Techniques
					EOR	Enhanced Oil Recovery
Ag.E	GWH	Ground Water Hydrology	LWME	Land and Water Management	DBS	Design of Bio-Energy Systems
	PHECC	Post-harvest Engineering of Cereal Crops	SWH	Surface Water Hydrology	PHEHP	Post-Harvest Engineering for Horticultural Produce
			MIE	Micro Irrigation Systems		
	FRES	Fundamentals of Renewable Energy Sources			AME	Agricultural Machinery and Equipment
					APEFQ	Agriculture Process Engineering and Food Quality
EDC	NPD	New Product Development	ESB	Entrepreneurship and Small Business Management	BPS	Business Policy & Strategic Management
			CIM	Change & Innovations Management	GM	Green Marketing
			PFP	Personal Financial Planning	SM	Startup Management
			EBM	E-Business Management	VM	Venture Management
			IRM	Insurance and Risk Management		

Major Core Courses (MCC)

Course Code	Course Name	Level	L	T	P	C	CIE	SEE	Total	Pre-requisite
2501MA01	Linear Algebra & Calculus	FC	2	1		3	50	50	100	-
2501MA02	Differential Equations & Vector Calculus	FC	2	1		3	50	50	100	-
2501CS71	Computer Organization & Architecture	FC	2	1		3	50	50	100	DLD
2501PH02	Modern Physics	FC	2		1	3	50	50	100	-
2501ME01	Engineering Graphics	FC	1		2	3	50	50	100	-
2501IT42	Programming with Python	FC	2		2	4	50	50	100	PPSC
2501CS01 / 2601DS01	Programming for Problem Solving Using C* / Problem Solving Through C Programming#	FC	2		2	4	50	50	100	-
2501CS03/ 2601DS02	Data Structures* / Data Structures using C++#	FC	2		2	4	50	50	100	PPSC
2501IT01/ 2601AI01	Business Intelligence Lab* / Business Intelligence & AI Tools Lab#	FC			2	2	50	50	100	-
2501MA08	Discrete Mathematics	IC	2	1		3	50	50	100	LAC
2501MA09	Probability & Statistics	IC	2	1		3	50	50	100	LAC
2501AI02	Artificial Intelligence	IC	2		1	3	50	50	100	-
2501CS08	Object Oriented Proramming through C++*	IC	2		2	4	50	50	100	PPSC
2501IT05	Database Management Systems	IC	2		2	4	50	50	100	PPSC
2501AI03	Data Mining	IC	2		1	3	50	50	100	-
2501IT06	Java Programming	IC	2		2	4	50	50	100	PPSC
2601AI02	Full Stack Application Development using MERN#	IC	2		2	4	50	50	100	JP
2501CS09	Language Processors	IC	2	1		3	50	50	100	PPSC
2501IT07	Agile Software Engineering	IC	2		1	3	50	50	100	PPSC
2501CS13	Operating Systems	IC	2		1	3	50	50	100	-
2501CS07	Computer Networks	IC	2		1	3	50	50	100	-
2501AI05	Machine Learning	IC	2		2	4	50	50	100	DAE
2501CS10	Advanced Data Structures & Algorithm Analysis	AC	2		1	3	50	50	100	DS
2501AI11	Deep Learning	AC	2		2	4	50	50	100	DAE
2501AI20	Natural Language Processing & Prompt Engineering	AC	2		1	3	50	50	100	ML

2501AI21	Reinforcement Learning & Gen AI	AC	2		1	3	50	50	100	ML
2501AI04	Big Data Analytics	AC	2		1	3	50	50	100	DM
Total			**51**	**6**	**32**	**89**				

Linear Algebra & Calculus

(Common to CE, EEE, ME, ECE, CSE, IT, AIML, CSE(DS), PT & Min.E)

	L	T	P	C
Course Code: 2501MA01	**2**	**1**	**0**	**3**

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

CO1: Solve the system of Linear equations
CO2: Calculate Eigen values and Eigen vectors
CO3: Apply differential calculus for one and several variable functions
CO4: Calculate the Maximum value and Minimum value of a function of several variables
CO5: Compute areas and volumes using multiple integrals

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	3	2	-	-	-	-	-	-	-	-	-
CO2	3	2	-	-	-	-	-	-	-	-	-
CO3	3	2	-	-	-	-	-	-	-	-	-
CO4	3	2	-	-	-	-	-	-	-	-	-
CO5	3	2	-	-	-	-	-	-	-	-	-

UNIT-I

System of linear equations: Vector Space,Linear Independence,Rank of a matrix by echelon form, normal form, Inverse of Non-singular matrices by Gauss-Jordan method, Solutions of Linear Systems: Existence, Uniqueness,Solving the system by Gauss elimination method.

Practice(Using any computational tool):

1. Variables, arithmetic operations, elementary mathematical functions.
2. Defining row vector, column vector, Arithmetic operations on matrices
3. finding transpose of a matrix, inverse of a matrix, determinant of a matrix
4. rank of a matrix, , solving system of linear equations.

UNIT-II

Eigenvalues, Eigvectors : Eigenvalues and properties(without proof), Eigenvectors, Diagonalization of a matrix, Cayley-Hamilton Theorem (without proof), Quadratic forms, Reduction of Quadratic form to canonical forms by Orthogonal Transformation,Nature of Quadratic forms.

Practice(Using any computational tool):

1. Computing eigen values and eigen vectors, matrix diagonalization..

UNIT-III

One Variable Calculus:Cauchy's mean value theorem, Taylor's and Maclaurin theorems with remainders (without proof), Problems and applications on the above theorems.
Several Variable Calculus: Limit, Continuity, partial derivatives and their geometrical interpretation.

Practice(Using any computational tool):

1. Basics of plotting,Plot graphs of single variable functions

UNIT-IV

Functions of several variables: Total differential and differentiability,derivatives of composite and implicit functions,derivatives of higher order and their commutativity,Euler's theorem on homogeneous functions , Taylor's and Maclaurin's expansion of functions of two variables. Jacobians, maxima and minima, constrained maxima/minima problems using Lagrange's method of multipliers.

Practice(Using any computational tool):

1. Plot graphs of various multi variable functions.

UNIT – V

Multiple Integrals: Double integrals, triple integrals, change of order of integration, change of variables to polar, cylindrical and spherical coordinates. Finding areas (by double integrals) and volumes (by double integrals and triple integrals).

Practice(Using any computational tool):

1. Plotting the region of Integration

Students are advised to use any computational / AI Tool like Wolfrum Alpha, Symbolab, Mathway, Desmos, Geogebra etc., for the practice

Text Books:

1. Advanced Engineering Mathematics, R. K. Jain and S. R. K. Iyengar, Alpha Science International Ltd., 5th Edition (9th reprint), 2021· ISBN 978-8184875607
2. Advanced Modern Engineering Mathematics, Glyn James, Pearson publishers, 5th Edition, 2018. ISBN-13. 978-1292174341

Reference Books:

1. Advanced Engineering Mathematics, Michael Greenberg, Pearson publishers, 9th edition. ISBN-13. 9788177585469
2. Higher Engineering Mathematics, H. K. Dass, Er. R. Verma, S-Chand publishers, 3rd edition 2023.ISBN 9788121938907

Web Links:

1 https://archive.nptel.ac.in/courses/111/104/111104137/
2 https://archive.nptel.ac.in/courses/111/107/111107108/
3 https://www.khanacademy.org/math/linear-algebra/
4 https://www.khanacademy.org/math/multivariable-calculus

Differential Equations & Vector Calculus
(Common to CE, EEE, ME, ECE, CSE, IT, AIML, CSE(DS), PT & Min.E)

	L	T	P	C
Course Code: 2501MA02	**2**	**1**	**0**	**3**

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

CO1: Solve the linear differential equations of first order and apply in various engineering problems.
CO2: Solve the linear differential equations of higher order and apply in various engineering problems.
CO3: Solve the linear partial differential equations.
CO4: Calculate the gradient, divergence and curl.
CO5: Compute work done, flux using vector integration.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	3	2	-	-	-	-	-	-	-	-	-
CO2	3	2	-	-	-	-	-	-	-	-	-
CO3	3	2	-	-	-	-	-	-	-	-	-
CO4	3	2	-	-	-	-	-	-	-	-	-
CO5	3	2	-	-	-	-	-	-	-	-	-

UNIT-I

Ordinary Differential Equations of First Order and First Degree: Solution of first order linear differential equations, exact differential equations and equations reducible to exact differential equations, Orthogonal Trajectories, Modelling of RL- circuit.

Practice(Using any computational tool) :

1. Solving the first order initial value problems using **odesolver** and plot the solution curves

UNIT-II

Linear Differential Equations of Higher Order: Solution of linear differential equations with constant coefficients, method of variation of parameters, solution of simultaneous linear differential equations.
Equations reducible to Linear differential equations with constant coefficients: Cauchy's homogeneous Linear Equations, Legendere's Linear Equations. Study of oscillations arising in LCR circuit (free oscillations and forced oscillations).

Practice(Using any computational tool) :

1. Solving the second order initial value problems using **odesolver** and plot the solution curves.

UNIT-III

Partial Differential Equations: Solution of linear PDE of first order by Lagrange's method, solution of homogeneous linear PDE of higher order with constant coefficients.

UNIT-IV

Vector Differentiation: Gradient of a scalar field, finding angle between two surfaces, directional derivative. Divergence and solenoidal fields. Curl and irrotational fields, Finding Scalar Potential.

Practice(Using any computational tool) :

1. Plotting of surfaces, 3D-plots, plotting vector fields.

UNIT – V

Vector Integration: Line integrals, work done by a force, conservative force field, surface integral, flux, volume integral. Green's theorem, Stoke's theorem and Gauss divergence theorem.

Students are advised to use any computational / AI Tool like Wolfrum Alpha, Symbolab, Mathway, Desmos, Geogebra etc., for the practice

Text Books:

1 Advanced Engineering Mathematics, E. Kreyszig, John Willey & Sons,10th Ed., 2018. ISBN 978-0470458365

2 Higher Engineering Mathematics, B. S. Grewal, Khanna Publishers, 44'th Edition (2021). ISBN 978-9383214204

Reference Books:

1 Advanced Engineering Mathematics, Dennis G. Zill, Jones & Bartlett Learning, 2018, 6th Edition.ISBN 978-1284105902.

2 Higher Engineering Mathematics, B.V. Ramana, McGraw-Hill Education, 11'th Ed., 2017. ISBN 978-9339216016.

Web Links:

1 https://www.classcentral.com/course/differential-equations-engineers-13258
2 https://archive.nptel.ac.in/courses/111/106/111106100/
3 https://www.khanacademy.org/math/differential-equations
4 https://archive.nptel.ac.in/courses/111/101/111101153/
5 https://archive.nptel.ac.in/courses/111/105/111105122/

Computer Organization & Architecture

(Common to CSE, IT, AIML & CSE (DS))

	L	T	P	C
Course Code: 2501CS71	**2**	**1**	**0**	**3**

Course Outcomes:

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

- **CO1:** Describe the basic structure of a computer system, various number systems and arithmetic operations
- **CO2:** Explain the Operation of CPUs including RTL, ALU, Instruction Cycle and Buses
- **CO3:** Demonstrate the architecture and functionality of central processing unit
- **CO4:** Illustrate the I/O and memory organization in an efficient way.
- **CO5:** Make use of multi processors and pipelining to improve the efficiency of computer system.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	2	1	-	-	2	-	-	-	-	-	-
CO2	2	1	-	-	2	-	-	-	-	-	-
CO3	3	1	-	-	2	-	-	-	-	-	-
CO4	2	1	-	-	2	-	-	-	-	-	-
CO5	3	2	-	-	3	-	-	-	-	-	-

Mapping of Course Outcomes with Program Specific Outcomes:

CO / PSO	PSO 1	PSO 2
CO1	2	-
CO2	2	-
CO3	2	-
CO4	2	-
CO5	3	-

UNIT-I

Basic Structure of Computers: Basic Organization of Computers, Historical Perspective, Bus Structures. Data Representation: Data types, Complements, Fixed Point Representation. Floating – Point Representation. Other Binary Codes, Error Detection Codes.

Computer Arithmetic: Addition and Subtraction, Multiplication Algorithms, Division Algorithms

UNIT-II

Register Transfer Language and Microoperations: Register Transfer language. Register Transfer Bus and Memory Transfers, Arithmetic Micro operations, Logic Micro

Operations, Shift Micro Operations, Arithmetic Logic Shift Unit

Basic Computer Organization and Design: Instruction Codes, Computer Register, Computer Instructions, Instruction Cycle, Memory – Reference Instructions. Input – Output and Interrupt, Complete Computer Description

UNIT-III

Central Processing Unit: General Register Organization, STACK Organization. Instruction Formats, Addressing Modes, Data Transfer and Manipulation, Program Control, Reduced Instruction Set Computer.
Microprogrammed Control: Control Memory, Address Sequencing, Micro Program example, Design of Control Unit.

UNIT-IV

Memory Organization: Memory Hierarchy, Main Memory, Auxiliary Memory, Associative Memory, Cache Memory, Virtual Memory.
Input-Output Organization: Peripheral Devices, Input-Output Interface, Asynchronous data transfer, Modes of Transfer, Priority Interrupts, Direct Memory Access

UNIT – V

Multi Processors: Introduction, Characteristics of Multiprocessors, Interconnection Structures, Inter Processor Arbitration.
Pipeline: Parallel Processing, Pipelining, Instruction Pipeline, RISC Pipeline, Array Processor

Text Books:

1 Computer System Architecture, M. Morris Mano, Third Edition, Pearson, 2008.
2 Computer Organization, Carl Hamacher, ZvonkoVranesic, SafwatZaky, 5/e, McGraw Hill, 2002.

Reference Books:

1 Computer Organization and Architecture, William Stallings, 6/e, Pearson, 2006.
2 Structured Computer Organization, Andrew S. Tanenbaum, 4/e, Pearson, 2005.
3 Fundamentals of Computer Organization and Design, Sivarama P. Dandamudi, Springer, 2006

Web Links:

1 https://nptel.ac.in/courses/106/105/106105163/
2 https://nptel.ac.in/courses/106/106/106106092/
3 https://www.udemy.com/course/computer-architecture-computer-organization-course/
4 http://www.cuc.ucc.ie/CS1101/David%20Tarnoff.pdf

Modern Physics

(Common to EEE, ECE, CSE, IT, AIML & CSE (DS))

	L	T	P	C
Course Code: 2501PH02	**2**	**0**	**1**	**3**

Course Outcomes:

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

- **CO1:** Apply the principles of interference and diffraction to design and enhance the resolving power of grating
- **CO2:** Familiarize the concepts of LASER's and Optical fibres with applications.
- **CO3:** Explain the fundamental concepts of Quantum behavior of matter
- **CO4:** Differentiate various electron theories to understand the properties of solids
- **CO5:** Explain the basic concepts of Semiconductors and identify the type of semiconductors using Hall effect.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	2	1	-	-	-	-	-	1	1	-	-
CO2	2	1	-	-	-	-	-	1	1	-	-
CO3	2	1	-	-	-	-	-	1	1	-	-
CO4	2	1	-	-	-	-	-	1	1	-	-
CO5	2	1	-	-	-	-	-	1	1	-	-

UNIT-I

Wave Optics

Interference: Introduction - Principle of Superposition - Interference of light- Conditions for sustained Interference**-** Interference in thin films (reflected geometry) – Colours in thin films – Newton's Rings (reflected geometry) – Determination of wavelength and refractive index – Applications.

Diffraction: Introduction – Fresnel and Fraunhofer Diffraction - Diffraction due to Single slit (quantitative), Double slit (qualitative) - N–Slits Intensity distribution curves (qualitative) – Grating spectrum–Rayleigh's criterion - Resolving powers of grating (qualitative) - Applications.

Practice:

1. Determination of Radius of curvature of a plano convex lens using Newton's Rings.
2. Measurement of Width or thickness of a thin wire by forming interference fringes.
3. Determination of Wavelength of light source using Diffraction grating by normal incidence method.
4. Resolving power of grating.
5. Determination of wavelength of light by using prism

UNIT-II

Lasers & Optical Fibres

Lasers: Introduction - Characteristics of Lasers – Spontaneous and Stimulated emission of radiation – population inversion- Einstein's coefficients – Relation - significance - Pumping Mechanisms - Ruby laser – Helium-Neon laser - Applications.

Optical Fibers: Introduction – Principle of propagation in Optical Fiber – Angle of acceptance – Expression for Numerical Aperture and condition for propagation – Classification of Optical fibers - Applications.

Practice:

1. Determination of Wavelength of He-Ne laser source by using diffraction grating.

UNIT-III

Quantum Mechanics Introduction – Matter waves – de Broglie's hypothesis – Davisson and Germer Experiment - Heisenberg's Uncertainty Principle – interpretation of wave function – Schrödinger Time dependent and Time Independent wave equations– Particle in a potential box.

UNIT-IV

Free electron Theory: Introduction–Classical free electron theory (merits and demerits only) -Quantum Free electron theory – Electrical conductivity- Fermi energy state–Fermi Dirac distribution function - Temperature dependence – Density of states.

Band theory of solids: Bloch Theorem – Origin of energy bands in crystalline solids – classification of crystalline solids.

Practice:

1. Study the variation of magnetic field along the axis of a circular coil carrying current by using Stewart and Gee's apparatus.
2. Determination of Frequency of electrically maintained tuning fork by Melde's apparatus.

UNIT – V

Semiconductor Physics Introduction – Intrinsic semiconductors – density of charge carriers – Electrical conductivity –Fermi level - extrinsic semiconductors - P-type & N-type semiconductors-Density of charge carriers (Qualitative) - Dependence of Fermi energy on carrier concentration and temperature–Hall effect-Hall coefficient - Applications of Hall effect–Drift and Diffusion currents - Einstein's equation

Semiconductor Devices Working of PN junction diode – Zener diode Forward and reverse bias Zener diode -Metal-Oxide-Semiconductor (MOS) structure- Capacitance-voltage characteristics-MOSFET structure - I–V characteristics.

Practice:

1. Determination of V-I characteristics and Breakdown voltage of a Zener diode.
2. Determination of Energy band gap of a semiconductor by using P-N junction diode.
3. Study the relation between Temperature and resistance and finding the constants A

& B of a thermistor.

4. Determine the resistivity of a semiconductor by four probe method.

Text Books:

1 Engineering Physics by M N Avadhanulu & T.V.S. Arun Murthy, S Chand &Company Ltd, 1st edition 2024. ISBN: 978-93-5870-932-3

2 Engineering Physics" by Satyendra Sharma and Jyotshna Sharma, Pearson publications, ISBN: 978-81-3151-178-7

Reference Books:

1 Concepts of Modern Physics by Authur Beiser, Shobhit Mahajan and S Rai Choudhary, McGraw Hill, ISBN: 9789351341857

2 Engineering Physics by M.R. Srinivasan, New Age international publishers, ISBN: 978-1848290501

3 "Optics" by Ajoy Ghatak, McGraw Hill Education, 6th Edition, ISBN: 978-9390113590

Web Links:

1 http://nptel.ac.in/courses/122107035/11

2 http://nptel.ac.in/courses/115102023/-

3 https://phet.colorado.edu/en/simulations/category/physics

4 http://physicsgecg.blogspot.in/p/reading-materials.html

Engineering Graphics

(Common to CE, EEE, ME, ECE, CSE, IT, AIML, CSE(DS), PT & Min.E)

	L	T	P	C
Course Code: 2501ME01	**1**	**0**	**2**	**3**

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

CO1: Apply the principles of engineering drawing to construct Engineering curves
CO2: Construct projections of points and lines.
CO3: Demonstrate visualization skills of projections of planes.
CO4: Demonstrate visualization skills of projections of solids and development of surfaces
CO5: Construct isometric and orthographic views of simple solids.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	3	2	1	-	2	-	-	-	1	1	-
CO2	3	2	1	-	2	-	-	-	1	1	-
CO3	3	2	1	-	2	-	-	-	1	1	-
CO4	3	2	1	-	2	-	-	-	1	1	-
CO5	3	2	1	-	2	-	-	-	1	1	-

Practice:

1. **Introduction to Engineering Graphics**
   a. Introduction to AutoCAD, Dimensioning, elements of dimensioning, system of dimensioning, Conventions in Drawing.

2. **Construction of Cycloids and Involutes**
   a. Construction of Cycloid, Epicycloid and Hypocycloid
   b. Involute of a pentagon and circle

3. **Introduction to Orthographic projection**
   a. Principles of Projection
   b. Orthographic Projection – Four Quadrants
   c. First angle & Third angle Projection with examples Reference plane, importance of reference lines or Plane
   d. Projections of a point situated in any one of the four quadrants

4. **Projection of straight lines-I**
   a. Projections of straight lines parallel to both reference planes.
   b. Projections of straight lines perpendicular to one reference plane and parallel to other reference plane
   c. Projections of straight line parallel to one plane & inclined to another plane

5. **Projection of straight lines-II**
   a. Projections of straight line inclined to both reference planes

6. **Projection of planes**
    a. Regular planes perpendicular to both reference planes, Parallel to one reference plane and inclined to the other reference plane
    Ex: Rectangle, Pentagon, Hexagon and Rhombus.

7. **Projections of Planes**
    a. Projections of Planes inclined to both reference planes Ex: Rectangle, Pentagon, Hexagon and Rhombus etc.

8. **Projection of solids**
    a. Axis Perpendicular to H.P and Axis Perpendicular to V.P
    Ex: Pentagonal and Hexagonal Prisms, Pyramids, Cylinder and Cone
    b. Axis Parallel to H.P and V.P
    Ex: Pentagonal and Hexagonal Prisms, Pyramids, Cylinder and Cone

9. **Projection of Solids**
    a. Projection of Solids with axis inclined to one reference plane and parallel to another plane
    Ex: Pentagonal and Hexagonal Prisms, Pyramids, Cylinder and Cone

10. **Development of Surfaces**
    a. Development of a Prisms and Cylinder simple cases

11. **Development of Surfaces**
    a. Development of Pyramids and Cone simple cases

12. **Conversion of Isometric views to Orthographic views**
    a. Practice figure -1
    b. Practice figure -2

Additional Practice:

1. **Conversion of Isometric views to Orthographic views**
    a. Practice figure – 3
    b. Practice figure - 4
2. **Conversion of Orthographic views to Isometric views**
    a. Practice figure - 1
    b. Practice figure - 2

Text Books:

1 Engineering Drawing, N. D. Bhatt, Charotar Publishing House, 54th edition, 2024, ISBN : 9789385039706.

2 Engineering Drawing and Graphics , Venugopal, New Age Publications, 2nd edition, 2019, ISBN: 9788122415452.

Reference Books:

1 Engineering Drawing, K.L. Narayana and P. Kannaiah, Tata McGraw Hill, 2021, ISBN: 978-9385983177.

2 Computer Aided Engineering Graphics, T. Jeyapoovan,Vikas Publishing house, New Delhi, 1stEdition, 2023, ISBN : 9789356743199.

Web Links:

1 https://nptel.ac.in/courses/112103019/

2 https://academy.autodesk.com/authenticated-home-user

3 https://www.sciencedirect.com/book/9780080108391/engineering-drawing-from-thebeginning

Programming with Python
(Common to CSE, IT, AI&ML and DS)

	L	T	P	C
Course Code: 2501IT42	**2**	**0**	**2**	**4**

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

CO1: Develop Python programs using basic programming constructs, control statements, loops.
CO2: Implement data processing solutions using Python's string methods and core data structures.
CO3: Develop real time applications using functions and packages.
CO4: Apply Object Oriented Programming concepts and files.
CO5: Build various applications using GUI and exceptions.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	2	2	3	-	2	-	-	1	-	-	1
CO2	2	2	3	-	2	-	-	1	-	-	1
CO3	3	-	2	-	2	-	-	2	1	1	1
CO4	2	1	2	-	2	-	-	1	1	1	1
CO5	1	1	2	-	2	-	-	2	1	1	1

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1	2	2
CO2	3	2
CO3	2	2
CO4	2	2
CO5	1	2

UNIT – I

Programming:
Introduction to Programming, Introduction to Python, Program Development Cycle, Input–Processing–Output, Print Function, Comments, Variables, Keyboard Input, Basic Calculations, Operators, Type Conversions, Expressions, Data Types, Strings, Using Functions, Using Modules. Decision Making: if, if-else, if-elif-else, Nested Decisions, Logical Operators, Boolean Expressions. Loops: while, for, Running Totals, Input Validation, Nested Loops.

Practice:

1. Basic Programs.

a. Adding two numbers

https://www.codechef.com/problems/FLOW001

b. Perfect Square root or not

c. Perfect Number

d. Even Number or Odd number

https://www.hackerrank.com/challenges/py-if-else/problem?isFullScreen=true

2. Decision Structures and Loops.

a. Python Loops – Printing Squares of Numbers
https://www.hackerrank.com/challenges/python-loops/problem?isFullScreen=true
b. Check for Armstrong number, Palindrome.
https://www.hackerrank.com/contests/iiitdm-contest1/challenges/armstrong-number-2?utm
c. Spy Number
d. Compound Interest
e. Count vowels, consonants, digits in a string.
https://www.codechef.com/problems/CV

f. Print a number pyramid.
https://www.codechef.com/problems/PATT01

UNIT – II

Python Data Types:

Working with Strings: Basic String Operations, String Formatting, String Methods. Lists and Tuples: Sequence Operations, Lists, Tuples, Tuple Function. Dictionaries: Creating and Using Dictionaries, Basic Dictionary Operations, Dictionary Methods. Sets: Creating Sets, Basic Set Operations, Set Methods, Simple Set Comprehensions.

Practice:

a. String Matching in an Array

https://leetcode.com/problems/string-matching-in-an-array/

b. Product-of-Three-Numbers

https://leetcode.com/problems/maximum-product-of-three-numbers

c. Write a program to check if the substring is present in a given string or not.

https://www.hackerrank.com/challenges/find-a-string/problem?utm

d. Reverse-vowels-of-a-String

https://leetcode.com/problems/reverse-vowels-of-a-string

Lists, Tuples, Dictionary

a. Generate 20 random numbers; print min, max, avg, even count.
https://www.codechef.com/problems/FLOW002

b. **Longest Consecutive Sequence**

https://leetcode.com/problems/longest-consecutive-sequence/

c. **Subarray Sum Equals K**

https://leetcode.com/problems/subarray-sum-equals-k/

UNIT – III

Functions and Packages: Function Definition, Parameters, Local vs Global Variables, Default Arguments, Keyword Arguments, Return Statement, Lambda Functions. Modules: import, from–import, Creating Simple Modules, Standard Modules, Packages. Basic Problem Solving: Top-Down Design, Simple Recursion Concepts.

Practice:

1. Working with Functions

a. Write a Python program to define a function that takes parameters and returns the sum of two numbers.

https://www.geeksforgeeks.org/python-program-to-find-factorial-of-number-using-recursion/

b. **Two Sum** — a classic easy problem that's often solved using a function.
https://leetcode.com/problems/two-sum/

c. **Valid Anagram** — good practice for string/dictionary manipulations inside a function.
https://leetcode.com/problems/valid-anagram/

d. **FLOW001** — "Add Two Numbers": simplest problem, good for writing a function to sum two numbers. https://www.codechef.com/problems/FLOW001

e. **FLOW002** — "Compute the Average": practice writing a function to compute averages.
https://www.codechef.com/problems/FLOW002

UNIT – IV

Object Oriented Programming: Concept of class, object and instances, Constructor, class attributes and destructors, Real time use of class in live projects, Inheritance , overlapping and overloading operators, Structuring Classes with Inheritance and Polymorphism File Operations: Reading config files in python, Writing log files in python, Understanding read functions, read(), readline() and readlines(), Understanding write functions, write() and writelines(), Manipulating file pointer using seek, Programming using file operations.

Practice:

1. OOP – Classes and Methods

a. Create a Product class with dynamic pricing and stock update.
https://www.codechef.com/problems/STFOOD

b. Define Converter class for unit conversions (e.g., inches → feet).
https://www.codechef.com/problems/CONV01

c. Create a Time class to convert seconds → minutes, hours.
https://www.codechef.com/problems/TIMEMNG

2. File Handling
   a. **Say "Hello, World!" from a file** (read input from file, print output): https://www.hackerrank.com/challenges/write-a-function/problem
   b. **Reading from STDIN / file and processing lines**: https://www.hackerrank.com/challenges/whats-your-name/problem
   c. **FLOW001 – Add Two Numbers**: https://www.codechef.com/problems/FLOW001

UNIT – V

Errors and Exceptions: Syntax Errors, Exceptions, Handling Exceptions, Raising Exceptions, User- defined Exceptions, Defining Clean-up Actions, Redefined Clean up Actions. Graphical User Interfaces: The Behaviour of Terminal Based Programs and GUI - Based, Programs, Coding Simple GUI-Based Programs.

Practice:

1. GUI and Exceptions
   a. Build a file-open dialog and display content in a text box. https://www.codechef.com/problems/ERROR404
   b. **Python Try/Except** — basic problem to handle division by zero and other exceptions: https://www.hackerrank.com/challenges/py-exceptions/problem
   c. **Handling Multiple Exceptions** — practise catching multiple exception types in one block: https://www.hackerrank.com/challenges/handling-multiple-exceptions/problem
   d. Show file handling using try/finally and with statements. https://www.codechef.com/problems/VALIDMIN

2. Integrated Problem Solving
   a. Implement the Hangman game using string logic and loops. https://www.codechef.com/problems/STRPALIN

Text Books:

1 Fundamentals of Python First Programs, Kenneth. A. Lambert, Cengage, 2018.
2 Beginning Python: from Novice to Professional, Lie Hetland, Magnus, 2nd Edition, 2005.

Reference Books:

1. Introduction to Python Programming, Gowrishankar.S, Veena A, CRC Press.
2. Introduction to Programming Using Python, Y. Daniel Liang, Pearson, 2017.
3. Think Python, Allen Downey, Green Tea Press,2012.
4. Python for Everybody Exploring Data in Python 3, Charles Russell Severance, Sue Blumenberg.

Web Links:

1 https://www.python.org/
2 https://www.coursera.org/courses?query=Python%20programming
3 https://www.learnpython.org/
4 https://onlinecourses.nptel.ac.in/noc21_cs32/preview/

Additional Practice:

LPYAS10 – *Print the Pattern*	https://www.codechef.com/practice/course/python/LPPYAS01/problems/LPYAS10
LPYAS09 – *Simple Input Output*	https://www.codechef.com/practice/course/python/LPPYAS01/problems/LPYAS09
LPYAS11 – *Check Even or Odd*	https://www.codechef.com/practice/course/python/LPPYAS02/problems/LPYAS11
LPYAS19 – *Sum of Digits*	https://www.codechef.com/practice/course/python/LPPYAS02/problems/LPYAS19
REMMON – *Remaining Monsters*	https://www.codechef.com/problems/REMMON
EPISODES – *Episodes*	https://www.codechef.com/problems/EPISODES
TRIANGLE7 – *Triangle Type*	https://www.codechef.com/problems/TRIANGLE7
COLORB – *Color Blindness*	https://www.codechef.com/problems/COLORB
CABRIDE – *Cab Ride*	https://www.codechef.com/problems/CABRIDE
SKONG – *Strong Language*	https://www.codechef.com/problems/SKONG
SELLCOIN – *Sell Coins*	https://www.codechef.com/problems/SELLCOIN
MIXINGLIQ – *Mixing Liquids*	https://www.codechef.com/problems/MIXINGLIQ
GAMBLING – *Gambling*	https://www.codechef.com/problems/GAMBLING
INCAT – *Increase the Number*	https://www.codechef.com/problems/INCAT
INDDAY – *Independence Day Offer*	https://www.codechef.com/problems/INDDAY
CWCTH – *Count the Holidays*	https://www.codechef.com/problems/CWCTH
Palindrome Number	https://leetcode.com/problems/palindrome-number
Valid Perfect Square	https://leetcode.com/problems/valid-perfect-square

Count the Digits That Divide a Number	https://leetcode.com/problems/count-the-digits-that-divide-a-number
Subtract the Product and Sum of Digits of an Integer	https://leetcode.com/problems/subtract-the-product-and-sum-of-digits-of-an-integer
Three Divisors	https://leetcode.com/problems/three-divisors
Smallest Even Multiple	https://leetcode.com/problems/smallest-even-multiple
Happy Number	https://leetcode.com/problems/happy-number
Ugly Number	https://leetcode.com/problems/ugly-number
Power of Two	https://leetcode.com/problems/power-of-two
Power of Three	https://leetcode.com/problems/power-of-three
Power of Four	https://leetcode.com/problems/power-of-four
Arranging Coins	https://leetcode.com/problems/arranging-coins
Excel Sheet Column Title	https://leetcode.com/problems/excel-sheet-column-title
Excel Sheet Column Number	https://leetcode.com/problems/excel-sheet-column-number

	Code chef	**Hacker rank**	**Leet code**	**Geeksfor Geeks**	**Total**
Unit-1	3	3	0	0	**6**
Unit-2	1	1	5	0	**7**
Unit-3	2	0	2	1	**5**
Unit-4	4	2	0	0	**6**
Unit-5	3	2	0	0	**5**
Additional Programs	16	0	16	0	**32**
Total	**29**	**8**	**23**	**1**	**61**

Programming for Problem Solving using C
(Common to CE, EEE, ME, ECE, CSE, IT, AIML, CSE(DS), PT & Min.E)
(Syllabus for the batch admitted in the year 2025)

	L	T	P	C
Course Code: 2501CS01	**2**	**0**	**2**	**4**

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

CO1: Demonstrate basics of computer, algorithm and flow chart for problem solving.
CO2: Make use of an appropriate control structures to solve given problems.
CO3: Solve complex problems using arrays and strings.
CO4: Develop modular programming using functions.
CO5: Demonstrate dynamics memory allocations and file handling using file operations.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1:	2	3	2	1	3	-	-	2	-	-	1
CO2:	2	3	2	1	3	-	-	2	-	-	1
CO3:	2	3	2	1	3	-	1	2	-	-	1
CO4:	2	3	2	1	3	-	1	2	-	-	1
CO5:	2	3	2	1	3	-	1	2	-	-	1

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1:	2	2
CO2:	2	2
CO3:	2	2
CO4:	2	2
CO5:	2	2

UNIT – I
Introduction to Programming and Problem Solving: Introduction to Programming Languages, Basics of a Computer Program- Algorithms, Algorithmic approach, characteristics of algorithm, Problem solving strategies: Top-down approach, Bottom-up approach, Time and space complexities of algorithms, flowcharts (Using Dia Tool), pseudo code.
Structure of C Program, Introduction to Compilation and Execution, Primitive Data Types, Variables, and Constants, Basic Input and Output, operators, keywords, identifiers, Type Conversion, and Casting.

Practice 1:

1.
   a. Basic linux environment and its editors like Vi, Vim & Emacs etc.
   b. Exposure to turbo C, gcc.
   c. Explore to hacker rank or any other Online coding platform and compiler environment.

d. "Hello world" in C.
e. Objective: Learn about the syntax of reading from stdin and writing to stdout.
https://www.hackerrank.com/challenges/hello-world-c/problem?isFullScreen=true
f. Write a simple program to read int, float, char and string using scanf() and display using printf() in all the above given platforms.

2. Basics and Operators
a. Sum and Difference of 2 numbers.
Objective: Learn int and float data types.
https://www.hackerrank.com/challenges/sum-numbers-c/problem?isFullScreen=true
b. Playing with Characters.
Objective: Learn how to take a character, a string and a sentence as input in C.
https://www.hackerrank.com/challenges/playing-with-characters/problem?isFullScreen=true
c. Bitwise Operators
Objective: Learn how to work with bits (0,1) and bitwise operators.
https://www.hackerrank.com/challenges/bitwise-operators-in-c/problem?isFullScreen=true
d. Conversion of Fahrenheit to Celsius and vice versa.
e. Distance travelled by an object.
f. Calculate Simple interest and compound interest.

3. Operators and Expressions, Variables and Type conversions.
a. Evaluate the following expressions
i. a/b*c-b+a*d/3
ii. j = (i++) + (++i)
b. Square root of a given number.
c. Find the area of circle, square, rectangle and triangle.
d. Find the maximum of three numbers using conditional operator.
e. Take marks of 5 subjects in integers, find the total in integer and average in float.

UNIT – II

Control Structures: Simple sequential programs, Conditional Statements (if, if-else, else if ladder, switch), Loops (for, nested for loop, while, do-while), break and continue, goto statement.

Practice:

1. Conditional Statements
a. Objective: Understand if and else Conditional statements in C.
https://www.hackerrank.com/challenges/conditional-statements-in-c/problem?isFullScreen=true
b. Roots of a Quadratic Equation.
c. Generate electricity bill.
d. Simulate a calculator using switch case.
e. Find the given year is a leap year or not.
2. Loops
a. Objective: Learn the usage of the for loop in C.
https://www.hackerrank.com/challenges/for-loop-in-c/problem?isFullScreen=true
b. Sum of the digits of a 5-digit number.
Objective: Learn the usage of while loop and usage of operators - % and /.
https://www.hackerrank.com/challenges/sum-of-digits-of-a-five-digit-

number/problem?isFullScreen=true
c. Given number is a prime or not. (Also Prime numbers between a given range.)
d. Armstrong Number or not.
e. Palindrome or not.
f. Objective: Print a pattern of numbers using Loops.
https://www.hackerrank.com/challenges/printing-pattern-2/problem?isFullScreen=true
g. Construct a Pyramid pattern.

UNIT – III

Arrays: Arrays indexing, Accessing programs with array of integers, two dimensional arrays, Introduction to Strings, string handling functions.
Sorting Techniques: bubble sort, selection sort.
Searching Techniques: linear, Binary search.

Practice:

1. 1-Demensional Arrays
   a. Objective: Print the sum and free the memory where the array is stored.
   https://www.hackerrank.com/challenges/1d-arrays-in-c/problem?isFullScreen=true
   b. Objective: Working with indices in array.
   https://www.hackerrank.com/challenges/reverse-array-c/problem?isFullScreen=true
   c. Search an element in array (Linear Search).
   d. Find min and max elements in array.
   e. Insert an element into array.
   f. Eliminate duplicate elements from array.
   g. Sorting of elements in an array using Bubble sort.
2. 2-Demensional Arrays
   a. Sum of two 2-D arrays.
   b. Multiplication of two 2-D arrays.
   c. Transpose of a Matrix.
   d. Trace of a Matrix.
   e. Lower Triangular Matrix.
3. Hacker Rank
   a. Objective: print each word of the sentence in a new line.
   https://www.hackerrank.com/challenges/printing-tokens- /problem?isFullScreen=true
   b. Count number of alphabets (lowercase, uppercase, consonants, vowels) and digits Lowercase to Uppercase, Uppercase to Lowercase, Toggle case, Sentential case
   c. Objective:Digit Frequency find the frequency of each digit in the given string.
   https://www.hackerrank.com/challenges/frequency-of-digits-1/problem?isFullScreen=true
   d. Find string length, concatenate 2 strings, reverse a string using built-in and without built-in string functions.

UNIT – IV

Functions: Introduction to Functions, Function Declaration and Definition, Function call Return Types and Arguments, arrays as parameters, Scope and Lifetime of Variables, storage classes, recursion, functions and arrays.

Practice: Functions in C

1. Objective: Learn simple usage of functions.

https://www.hackerrank.com/challenges/functions-in-c/problem?isFullScreen=true

2. Objective: Fibonacci Numbers using recursive function.
   https://www.hackerrank.com/challenges/ctci-fibonacci-numbers/problem
3. Objective: N^{th} factorial using recursion.
   https://www.hackerrank.com/contests/ccc-veltech-practice-set-ende/challenges/factorial-using-recursion-1
4. Objective: Find the super digit of the integer.
   https://www.hackerrank.com/challenges/recursive-digit-sum/problem
5. Implement LCM
6. Objective: Calculate the Nth term of series.
   https://www.hackerrank.com/challenges/recursion-in-c/problem?isFullScreen=true

UNIT – V

Pointer: Introduction to Pointers, dereferencing and address operators, pointer and address arithmetic, array manipulation using pointers, functions & pointers modifying parameters inside functions using pointers, Command line Arguments, Dynamic memory allocation, Null Pointer, generic pointer, dangling pointer.

File Handling: Introduction to Files, Using Files in C, Reading from Text Files, Writing to Text Files, Random File Access.

Practice:

1. Pointers
   a. Objective: learn to implement the basic functionalities of pointers in C.
      https://www.hackerrank.com/challenges/pointer-in-c/problem?isFullScreen=true
   b. Objective: Learn using Pointers with Arrays and Functions.
      https://www.hackerrank.com/challenges/students-marks-sum/problem?isFullScreen=true
   c. Objective: sort a given array of strings into lexicographically increasing order or into an order in which the string with the lowest length appears first.
      https://www.hackerrank.com/challenges/sorting-array-of-strings/problem?isFullScreen=true
   d. Find the sum of a 1D array using malloc().
   e. Swap two numbers using functions and pointers - call by value and reference.
   f. Objective: Dynamic Handling requests by a Librarian to place the books in the shelves.
      https://www.hackerrank.com/challenges/dynamic-array-in-c/problem?isFullScreen=true

2. File handling concepts
   a. Write text into and read text from a file.
   b. Write text into and read text from a binary file using fread() and fwrite().
   c. Copy the contents of one file to another file.
   d. Merge two files into the third file using command-line arguments
   e. Find no. of lines, words and characters in a file.

Additional Practice:

1. Variadic functions in C
   Objective: Understanding variable number of arguments
   https://www.hackerrank.com/challenges/variadic-functions-in-c/problem?isFullScreen=true
2. Small Triangles, Large Triangles
   Objective: Print sorted by their areas
   https://www.hackerrank.com/challenges/small-triangles-large-

triangles/problem?isFullScreen=true

3. Permutations of Strings
   Objective: print all strings permutations in strict lexicographical order
   https://www.hackerrank.com/challenges/permutations-of-strings/problem?isFullScreen=true

Text Books:

1	Programming in C, Rema Theraja, Oxford, 2nd edition. ISBN 93-5497-9
2	"The C Programming Language", Brian W. Kernighan and Dennis M. Ritchie, Prentice-Hall. ISBN 13: 9780131103627

Reference Books:

1	Computing fundamentals and C Programming, Balagurusamy, E., McGraw-Hill Education. ISBN.No: 9352604172
2	Byron Gottfried, Schaum's Outline of Programming with C, McGraw-Hill. ISBN No. 0071367993
3	Let Us C Yashwanth,Kanetkar, Eighth edition, BPB Publications.ISBN No. 1934015253
4	Programming in C A-Practical Approach Ajay Mittal. Pearson Education.ISBN No. 9788131729342
5	R G Dromey How to Solve It by Computer (Prentice-Hall International Series in Computer Science.ISBN-13 : 978-0134340012

Web Links:

1	https://www.hackerrank.com/
2	https://onlinecourses.nptel.ac.in/noc22_cs40/preview
3	https://archive.nptel.ac.in/courses/106/104/106104128/

Problem Solving Through C Programming
(Common to CE, EEE, ME, ECE, CSE, AIML, CSE(DS), PT & Min.E)
(Revised Syllabus for batches admitted from 2026 Onwards)

	L	T	P	C
Course Code: 2601DS01	**2**	**0**	**2**	**4**

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

- **CO1:** Demonstrate basics of computer, algorithm and flow chart for problem solving.
- **CO2:** Make use of an appropriate control structures to solve given problems.
- **CO3:** Solve complex problems using arrays and strings.
- **CO4:** Develop modular programming using functions and dynamic memory allocation.
- **CO5:** Develop programs using structures and file handling operations.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	2	3	2	1	3	-	-	2	-	-	1
CO2	2	3	2	1	3	-	-	2	-	-	1
CO3	2	3	2	1	3	-	1	2	-	-	1
CO4	2	3	2	1	3	-	1	2	-	-	1
CO5	2	3	2	1	3	-	1	2	-	-	1

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1	2	2
CO2	2	2
CO3	2	2
CO4	2	2
CO5	2	2

UNIT – I

Introduction to Programming and Problem Solving

Introduction to Programming Languages, Basics of a Computer Program- Algorithms, characteristics of algorithm, flowcharts, pseudo code.

Structure of C Program, Introduction to Compilation and Execution, Primitive Data Types, Variables, and Constants, Basic Input and Output, operators, keywords, identifiers, Type Conversion, and Casting and operator precedence.

Practice 1:

1. Explore different platforms
   a. Basic linux environment and its editors like Vi, Vim & Emacs etc.
   b. Exposure to turbo C, gcc.
   c. Explore to hacker rank or any other Online coding platform and compiler environment.
   d. "Hello world" in C.
      https://www.codechef.com/learn/course/c/CDEVINTRO/problems/CDEV004B
   e. Objective: Learn about the syntax of reading from stdin and writing to stdout.
      https://www.hackerrank.com/challenges/hello-world-c/problem?isFullScreen=true
   f. Write a simple program to read int, float, char and string using scanf() and display using printf() in

all the above given platforms.

2. Basics and Operators
   a. Sum and Difference of 2 numbers.
      Objective: Learn int and float data types.
      https://www.hackerrank.com/challenges/sum-numbers-c/problem?isFullScreen=true
   b. Playing with Characters.
      Objective: Learn how to take a character, a string and a sentence as input in C.
      https://www.hackerrank.com/challenges/playing-with-characters/problem?isFullScreen=true
   c. Bitwise Operators
      Objective: Learn how to work with bits (0,1) and bitwise operators.
      https://www.hackerrank.com/challenges/bitwise-operators-in-c/problem?isFullScreen=true
   d. Conversion of Fahrenheit to Celsius and vice versa.
      https://www.codechef.com/practice/course/c/LPCAS03/problems/LCAS30
   e. Distance travelled by an object.
   f. Calculate Simple interest and compound interest.

3. Operators and Expressions, Variables and Type conversions.
   a. Evaluate the following expressions
      i. a/b*c-b+a*d/3
      ii. j = (i++) + (++i)
   b. Square root of a given number.
   c. Find the area of circle, square, rectangle and triangle.
      https://www.hackerrank.com/challenges/rectangle-area/problem?utm_source=chatgpt.com
   d. Find the maximum of three numbers using conditional operator.
   e. Take marks of 5 subjects in integers, find the total in integer and average in float.

UNIT – II

Control Structures Conditional Statements (if, if-else, else if ladder, switch), Loops (for, nested for loop, while, do-while), break and continue, goto statement.

Practice:

1. Conditional Statements
   a. Objective: Understand if and else Conditional statements in C.
      https://www.hackerrank.com/challenges/conditional-statements-in-c/problem?isFullScreen=true
   b. Roots of a Quadratic Equation.
   c. Generate electricity bill.
   d. Simulate a calculator using switch case.
   e. Find the given year is a leap year or not.
2. Loops
   a. Objective: Learn the usage of the for loop in C. https://www.hackerrank.com/challenges/for-loop-in-c/problem?isFullScreen=true
   b. Sum of the digits of a 5-digit number.
      Objective: Learn the usage of while loop and usage of operators - % and /.
      https://www.hackerrank.com/challenges/sum-of-digits-of-a-five-digit-number/problem?isFullScreen=true
   c. Given number is a prime or not. (Also Prime numbers between a given range.)
   d. Armstrong Number or not.
   e. Palindrome or not.
   f. Objective: Print a pattern of numbers using Loops.
      https://www.hackerrank.com/challenges/printing-pattern-2/problem?isFullScreen=true
   g. Construct a Pyramid pattern.
      https://www.codechef.com/practice/course/c/LPCAS02/problems/LCAS20

UNIT – III

Arrays: Definition, types, 1-D array(declaration, initialization, representation, accessing, input and output), 2-D(declaration, initialization, representation, accessing, input and output), searching (Linear search, Binary Search non recursive) and sorting(Bubble Sort, Selection Sort, Insertion Sort).
Introduction to Strings: declaration, initialization, representation, accessing, input and output, string handling functions.
Practice:

1. 1-Dimensional Arrays
   a. Objective: Print the sum and free the memory where the array is stored. https://www.hackerrank.com/challenges/1d-arrays-in-c/problem?isFullScreen=true
   b. Objective: Working with indices in array.
      Search an element in array .
   c. Find min and max elements in array.
   d. Insert an element into array.
   e. Eliminate duplicate elements from array.
   f. Sorting of elements in an array
2. 2-Demensional Arrays
   a. Sum of two 2-D arrays.
   b. Multiplication of two 2-D arrays.
   c. Transpose of a Matrix.
   d. Trace of a Matrix.
   e. Lower Triangular Matrix.
3. Strings
   a. Objective: print each word of the sentence in a new line.
   b. Count number of alphabets (lowercase, uppercase, consonants, vowels) and digits Lowercase to Uppercase, Uppercase to Lowercase, Toggle case, Sentential case
   c. find the frequency of each digit in the given string. https://www.hackerrank.com/challenges/frequency-of-digits-1/problem?isFullScreen=true
   d. Find string length, concatenate 2 strings, reverse a string using built-in and without built-in string functions.

UNIT – IV
Pointer: Introduction to Pointers, dereferencing and address operators, pointer arithmetic, array manipulation using pointers, Dynamic memory allocation, Null Pointer, generic pointer, dangling pointer, pointer to pointer.
Functions: Introduction to Functions, Function Declaration and Definition, Function call Return **Types and Arguments**, Scope and Lifetime of Variables (block scope, function scope, program scope and file scope), **storage classes**, **recursion**, functions and arrays, Command line Arguments, functions and pointers.

Practice:

1. **Functions in C**

a. Objective: Learn simple usage of functions. https://www.hackerrank.com/challenges/functions-in-c/problem?isFullScreen=true
b. Objective: Fibonacci Numbers using recursive function. https://www.hackerrank.com/challenges/ctci-fibonacci-numbers/problem
c. Objective: N^{th} factorial using recursion.
d. Objective: Find the super digit of the integer.
e. Implement LCM
f. Objective: Calculate the Nth term of series. https://www.hackerrank.com/challenges/recursion-in-c/problem?isFullScreen=true

2. **Pointers**
   a. Objective: learn to implement the basic functionalities of pointers in C. https://www.hackerrank.com/challenges/pointer-in-c/problem?isFullScreen=true

b. Objective: Learn using Pointers with Arrays and Functions.
https://www.hackerrank.com/challenges/students-marks-sum/problem?isFullScreen=true
c. Objective: sort a given array of strings into lexicographically increasing order or into an order in which the string with the lowest length appears first.
https://www.hackerrank.com/challenges/sorting-array-of-strings/problem?isFullScreen=true
d. Find the sum of a 1D array using malloc().
e. Swap two numbers using functions and pointers - call by value and reference.
f. Objective: Dynamic Handling requests by a Librarian to place the books in the shelves.
https://www.hackerrank.com/challenges/dynamic-array-in-c/problem?isFullScreen=true

UNIT – V

Structures and Unions: Introduction, Nested Structures, Arrays of Structures, Structures and Functions, Self-Referential Structures, structures and pointers. Unions, Enumerated Data Type - enum variables, Using Typedef keyword, Bit Fields.

File Handling:-Introduction to Files, Using Files in C, Reading from Text Files, Writing to Text Files, File Operations, Random File Access.

Practice:

a. Structures
   a. Write a C program to find the total, average of n students using structures
   b. Copy one structure variable to another structure of the same type.
   c. Read student name and marks from the command line and display the student details along with the total.

b. File handling concepts
   a. Write text into and read text from a file.
   b. Write text into and read text from a binary file using fread() and fwrite().
   c. Copy the contents of one file to another file.
   d. Merge two files into the third file using command-line arguments
   e. Find no. of lines, words and characters in a file.

Text Books:

1 Programming in C, Rema Theraja, Oxford, 2nd edition. ISBN 93-5497-9
2 "The C Programming Language", Brian W. Kernighan and Dennis M. Ritchie, Prentice-Hall. ISBN 13: 9780131103627

Reference Books:

1. Computing fundamentals and C Programming, Balagurusamy, E., McGraw-Hill Education. ISBN.No: 9352604172
2. Byron Gottfried, Schaum's Outline of Programming with C, McGraw-Hill. ISBN No. 0071367993
3. Let Us C Yashwanth,Kanetkar, Eighth edition, BPB Publications.ISBN No. 1934015253
4. Programming in C A-Practical Approach, Ajay Mittal. Pearson Education.ISBN No. 9788131729342
5. R G Dromey How to Solve It by Computer (Prentice-Hall International Series in Computer Science.ISBN-13 : 978-0134340012

Web Links:

1 https://www.hackerrank.com/
2 https://onlinecourses.nptel.ac.in/noc22_cs40/preview

3 https://archive.nptel.ac.in/courses/106/104/106104128/

Additional Practice:

S.No.	Difficulty	Problem Name	Link
1	Easy	Add Two Numbers	https://www.codechef.com/practice/course/cpp/PCPP05/problems/FLOW001
2	Easy	Enormous Input Test	https://www.codechef.com/practice/course/basic-programming-concepts/DIFF500/problems/INTEST
3	Easy	ATM	https://www.codechef.com/practice/course/1-star-difficulty-problems/DIFF1200/problems/ATM2
4	Easy	Number Mirror	https://www.codechef.com/practice/course/python/PPY01/problems/START01
5	Easy	Sum of Digits	https://www.codechef.com/practice/course/basic-math/BASICMATH/problems/FLOW006
6	Easy	Remainder	https://www.codechef.com/practice/course/basic-math/BASICMATH/problems/FLOW002
7	Easy	First and Last Digit	https://www.codechef.com/practice/course/basic-programming-concepts/DIFF500/problems/FLOW004
8	Easy	Reverse The Number	https://www.codechef.com/practice/course/logical-problems/DIFF800/problems/FLOW007
9	Easy	Life, the Universe, and Everything	https://staging.codechef.com/practice/course/cpp/PRACTICECPP2/problems/TEST
10	Easy	Chef and Operators	https://www.codechef.com/practice/course/logical-problems/DIFF800/problems/CHOPRT
11	Easy	Difference and Sum	https://www.codechef.com/problems/DIFFSUM
12	Easy	Second Largest	https://www.codechef.com/practice/course/arrays/ARRAYS/problems/LARGESECOND
13	Easy	Two Sum	https://leetcode.com/problems/two-sum/
14	Easy	Reverse Integer	https://leetcode.com/problems/reverse-integer/
15	Easy	Palindrome Number	https://leetcode.com/problems/palindrome-number/
16	Easy	Roman to Integer	https://leetcode.com/problems/roman-to-integer/
17	Easy	Remove Duplicates from Sorted Array	https://leetcode.com/problems/remove-duplicates-from-sorted-array/
18	Easy	Remove Element	https://leetcode.com/problems/remove-element/
19	Easy	Find the Index of the First Occurrence in a String	https://leetcode.com/problems/find-the-index-of-the-first-occurrence-in-a-string/
20	Easy	Length of Last Word	https://leetcode.com/problems/length-of-last-word/
21	Easy	Plus One	https://leetcode.com/problems/plus-one/
22	Easy	Merge Sorted Array	https://leetcode.com/problems/merge-sorted-array/
23	Easy	Majority Element	https://leetcode.com/problems/majority-element/
24	Easy	Single Number	https://leetcode.com/problems/single-number/
25	Easy	Move Zeroes	https://leetcode.com/problems/move-zeroes/
26	Easy	Missing Number	https://leetcode.com/problems/missing-number/

27	Easy	Contains Duplicate	https://leetcode.com/problems/contains-duplicate/
28	Easy	Valid Anagram	https://leetcode.com/problems/valid-anagram/
29	Easy	Valid Parentheses	https://leetcode.com/problems/valid-parentheses/
30	Easy	Fibonacci Number	https://leetcode.com/problems/fibonacci-number/
31	Easy	Climbing Stairs	https://leetcode.com/problems/climbing-stairs/
32	Easy	Binary Search	https://leetcode.com/problems/binary-search/
33	Easy	Number of 1 Bits	https://leetcode.com/problems/number-of-1-bits/
34	Easy	Hamming Weight	https://leetcode.com/problems/number-of-1-bits/description/
35	Easy	Power of Two	https://leetcode.com/problems/power-of-two/
36	Easy	Power of Three	https://leetcode.com/problems/power-of-three/
37	Easy	Same Tree	https://leetcode.com/problems/same-tree/
38	Easy	Maximum Subarray	https://leetcode.com/problems/maximum-subarray/
39	Medium	Chef and Notebooks	https://www.codechef.com/problems/CNOTE
40	Medium	Chef and Wildcard Matching	https://www.codechef.com/search?q=TWOSTR
41	Medium	Minimum Moves	https://www.codechef.com/search?q=SALARY
42	Medium	Lucky Four	https://www.codechef.com/search?q=LUCKFOUR
43	Medium	Maximum Difference	https://www.codechef.com/search?q=MAXDIFF
44	Medium	Uncle Johny	https://www.codechef.com/search?q=JOHNY
45	Medium	Chef and Strings	https://www.codechef.com/search?q=CHEFSTLT
46	Medium	Two vs Rib	https://www.codechef.com/search?q=TWOVRIB
47	Medium	Chef and Array	https://www.codechef.com/search?q=CHEFARRP
48	Medium	Add Two Numbers	https://leetcode.com/problems/add-two-numbers/
49	Medium	Longest Substring Without Repeating Characters	https://leetcode.com/problems/longest-substring-without-repeating-characters/
50	Medium	Longest Palindromic Substring	https://leetcode.com/problems/longest-palindromic-substring/
51	Medium	3Sum	https://leetcode.com/problems/3sum/
52	Medium	Container With Most Water	https://leetcode.com/problems/container-with-most-water/
53	Medium	Integer to Roman	https://leetcode.com/problems/integer-to-roman/
54	Medium	String to Integer (atoi)	https://leetcode.com/problems/string-to-integer-atoi/
55	Medium	Divide Two Integers	https://leetcode.com/problems/divide-two-integers/

56	Medium	Rotate Image	https://leetcode.com/problems/rotate-image/
57	Medium	Set Matrix Zeroes	https://leetcode.com/problems/set-matrix-zeroes/
58	Medium	Spiral Matrix	https://leetcode.com/problems/spiral-matrix/
59	Medium	Jump Game	https://leetcode.com/problems/jump-game/
60	Medium	Permutations	https://leetcode.com/problems/permutations/
61	Medium	Combination Sum	https://leetcode.com/problems/combination-sum/
62	Medium	Search in Rotated Sorted Array	https://leetcode.com/problems/search-in-rotated-sorted-array/
63	Medium	Group Anagrams	https://leetcode.com/problems/group-anagrams/
64	Medium	Sort Colors	https://leetcode.com/problems/sort-colors/
65	Medium	Subsets	https://leetcode.com/problems/subsets/
66	Medium	Reverse Linked List II	https://leetcode.com/problems/reverse-linked-list-ii/
67	Medium	Add Two Numbers II	https://leetcode.com/problems/add-two-numbers-ii/
68	Medium	Reorder List	https://leetcode.com/problems/reorder-list/
69	Medium	Linked List Cycle II	https://leetcode.com/problems/linked-list-cycle-ii/
70	Medium	K-th Largest Element in an Array	https://leetcode.com/problems/kth-largest-element-in-an-array/
71	Medium	Product of Array Except Self	https://leetcode.com/problems/product-of-array-except-self/
72	Medium	Word Search	https://leetcode.com/problems/word-search/
73	Medium	Unique Paths	https://leetcode.com/problems/unique-paths/
74	Medium	Minimum Path Sum	https://leetcode.com/problems/minimum-path-sum/
75	Medium	Temperature Converter - II	https://www.codechef.com/practice/course/c/LPCAS03/problems/LCAS30B
76	Medium	Reverse	https://www.hackerrank.com/challenges/reverse-array-c/problem?isFullScreen=true
77	Medium	Sentence in a new line	https://www.hackerrank.com/challenges/printing-tokens-/problem?isFullScreen=true
78	Hard	Median of Two Sorted Arrays	https://leetcode.com/problems/median-of-two-sorted-arrays/
79	Hard	Regular Expression Matching	https://leetcode.com/problems/regular-expression-matching/
80	Hard	Wildcard Matching	https://leetcode.com/problems/wildcard-matching/
81	Hard	Trapping Rain Water	https://leetcode.com/problems/trapping-rain-water/
82	Hard	First Missing Positive	https://leetcode.com/problems/first-missing-positive/
83	Hard	Longest Valid	https://leetcode.com/problems/longest-valid-parentheses/

		Parentheses	
84	Hard	Merge k Sorted Lists	https://leetcode.com/problems/merge-k-sorted-lists/
85	Hard	Reverse Nodes in k-Group	https://leetcode.com/problems/reverse-nodes-in-k-group/description/
86	Hard	Sudoku Solver	https://leetcode.com/problems/sudoku-solver/
87	Hard	N-Queens	https://leetcode.com/problems/n-queens/description/
88	Hard	Edit Distance	https://leetcode.com/problems/edit-distance/
89	Hard	Word Ladder II	https://leetcode.com/problems/word-ladder-ii/
90	Hard	Maximal Rectangle	https://leetcode.com/problems/maximal-rectangle/
91	Hard	Binary Tree Maximum Path Sum	https://leetcode.com/problems/binary-tree-maximum-path-sum/
92	Hard	Serialize and Deserialize Binary Tree	https://leetcode.com/problems/serialize-and-deserialize-binary-tree/
93	Hard	Variadic functions in C	https://www.hackerrank.com/challenges/variadic-functions-in-c/problem?isFullScreen=true
94	Hard	Small Triangles, Large Triangles	https://www.hackerrank.com/challenges/small-triangles-large-triangles/problem?isFullScreen=true
95	Hard	Permutations of Strings	https://www.hackerrank.com/challenges/permutations-of-strings/problem?isFullScreen=true

	Basic Programs	**Code chef**	**Hacker rank**	**Leet code**	**Total**
Unit-1	4	2	5	0	**11**
Unit-2	4	1	4	0	**09**
Unit-3	8	0	2	0	**10**
Unit-4	5	0	7	0	**12**
Unit-5	9	0	0	0	**9**
Additional Practice	0	22	5	68	**95**
Total	**26**	**29**	**23**	**68**	**146**

Data Structures
(Common to EEE, ECE, CSE, IT, AIML & CSE (DS))
(Syllabus for the batch admitted in the year 2025)

	L	T	P	C
Course Code: 2501CS03	**2**	**0**	**2**	**4**

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

CO1: Utilize structure, union to handle heterogeneous data.
CO2: Illustrate Time and Space complexities for different searching and sorting Algorithms
CO3: Demonstrate various operations on Linked Lists
CO4: Explain different operations on Stack and Queue with applications.
CO5: Demonstrate the importance and various operation on non-linear data structures

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	1	1	2	-	-	-	-	-	1	1	-
CO2	1	1	2	2	-	-	-	-	1	1	-
CO3	1	2	2	1	-	-	-	-	1	1	-
CO4	1	1	2	1	-	-	-	-	1	1	-
CO5	1	1	2	2	-	-	-	-	1	1	-

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1	3	-
CO2	2	-
CO3	2	-
CO4	2	-
CO5	2	-

UNIT-I

Structures and Unions: Introduction, Nested Structures, Arrays of Structures, Structures and Functions, Self-Referential Structures, Unions, Enumerated Data Type – enum variables, Using Typedef keyword, Bit Fields.
Data Structures: Introduction to Data Structures, Types of Data Structures.

Practice:
1. Write a C program to find the total, average of n students using structures
2. Copy one structure variable to another structure of the same type.
3. Read student name and marks from the command line and display the student details along with the total.

UNIT-II

Introduction to Linear Data Structures: Definition and importance of linear data

structures, Abstract data types (ADTs) and their implementation, Overview of time and space complexity analysis for linear data structures.
Sorting Techniques: Quick sort, Merge sort, Radix sort

Practice:
1. Implement Merge sort using arrays.
2. Implement Quick sort using arrays
3. Implement Radix Sort using arrays

UNIT-III

Linked Lists: Singly linked lists: representation and operations, doubly linked lists and its operations and circular linked lists and its operations, Comparing arrays and linked lists, Applications of linked lists.

Practice:
1. Single Linked List: Perform different operations in single linked list.
2. Reversing a single linked list
3. Perform different operations in double linked list
4. Circular Linked List

UNIT-IV

Stacks: Introduction to stacks: properties and operations, implementing stacks using arrays and linked lists, applications of stacks.
Queues: Introduction to queues: properties and operations, Types of Queues, implementing queues using arrays and linked lists, applications of queues.

Practice:
1. Stack and its operations using arrays and Linked List
2. Implement a program to evaluate a postfix expression.
3. Queue and its operations using arrays and Linked List
4. Implement Circular Queue using Arrays

UNIT – V

Non-linear Data Structures:
Trees: Definition of tree, Tree Terminology, types of trees, Binary tree traversals, Binary Search Tree – Insertion, Deletion.
Graphs: Definition and Terminology – Representation of Graphs-Adjacency Matrix and Linked list, Graph Traversals (BFT & DFT)

Practice:
1. Implement Binary search tree (BST).
2. Implement Binary search Tree (BST) Traversals.
3. Graph Traversal using Breadth First Search (BFS)
4. Graph Traversal using Depth First Search (DFS)

Additional Practice:

1. Using a structure for transporting some boxes through a tunnel
2. Delete duplicate-value nodes from a sorted linked list
3. Convert an infix expression into postfix expression
4. De Queue Implementation

Text Books:

1	Data Structures using C, Rema Theraja, Oxford University Press, 3rd edition.ISBN9789354979453
2	Data Structures, KV Sambasivarao, S Rama Sree, S.Chand.ISBN:9789358704730.

Reference Books:

1	The Algorithm Design Manual, Steven S. Skiena, Springer Publication, Second Edition. ISBN: 978-1849967204
2	Fundamentals of data structures in C, Ellis Horowitz, Sartaj Sahni, Susan Anderson-Freed, Silicon Press. Data Structures and Algorithms by Maganti Venkatesh, Naresh. ISBN: 9780929306407
3	Data Structures Using C,Reema Thareja, Oxford University Press, 2nd Edition. ISBN:978-0198099307

Web Links:

1	https://nptel.ac.in/courses/106102064
2	https://archive.nptel.ac.in/courses/106/105/106105225/
3	https://www.udemy.com/topic/datastructures/
4	https://www.coursera.org/specializations/data-structures-algorithms
5	https://ds1-iiith.vlabs.ac.in/List%20of%20experiments.html

Data Structures using C++

(Common to EEE, ECE, CSE, AIML & CSE (DS))

(Revised Syllabus for batches admitted from 2026 Onwards)

	L	T	P	C
Course Code: 2601DS02	**2**	**0**	**2**	**4**

Course Outcomes:

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

CO1: Apply C++ features for problem solving.
CO2: Apply various sorting techniques to solve computing problems.
CO3: Make use of linear data structures to solve real time problems.
CO4: Develop applications using Tree data structures.
CO5: Solve problems using Graph algorithms.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	1	1	1					1	1		
CO2	1	1	2	2				1	1		
CO3	1	2	2	1				1	1		
CO4	1	1	2	1				1	1		
CO5	1	1	2	2				1	1		

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1	2	-
CO2	2	-
CO3	2	-
CO4	2	-
CO5	2	-

UNIT – I

An Overview of C++: Structure of C++ program, Input and output in C++, Class, Objects,Members Functions, Access Specifiers, Introduction to Constructors and Destructor.

Overview of Standard Template Library: Introduction, STL Programming Model, Containers, Sequence Containers, Associative Containers, Algorithms, Iterators, Vectors, Lists, Maps.

Practice:

1. Find the roots of a quadratic equation.
2. Find factorial of a given number using recursion
3. Palindrome Number
   https://leetcode.com/problems/palindrome-number/description/
4. Class and Objects
   https://www.hackerrank.com/challenges/classes-objects/problem?isFullScreen=true

5. Program to illustrate the use of Constructors and Destructors.
6. C++ Class Templates
   https://www.hackerrank.com/challenges/cpp-class-template-specialization/problem?isFullScreen=true

UNIT – II

Data Structures: Introduction to Data Structures, Types of Data Structures.

Linear Data Structures: Definition and importance of linear data structures, Abstract data types (ADTs) and their implementation.

Linked Lists: Singly linked lists: Representation and operations, Doubly linked lists and its operations and Circular linked lists and its operations, applications of linked list

Practice:

1. Single Linked List: Perform different operations in single linked list.
   https://leetcode.com/problems/design-linked-list/description/?utm_source
2. Perform different operations in double linked list
   https://leetcode.com/problems/design-linked-list/?utm_source
3. Circular Linked List
   https://www.codechef.com/practice/course/linked-lists/LINKLISTF/problems/PREP58?utm_source
4. Reversing a Linked List

UNIT – III

Stacks: Introduction to stacks: properties and operations, implementing stacks using arrays and linked lists, applications of stacks.

Queues: Introduction to queues: properties and operations, Types of Queues, implementing queues using arrays and linked lists, applications of queues.

Practice:

1. Stack and its operations using arrays and Linked List
   https://www.hackerrank.com/challenges/balanced-brackets/problem

   https://leetcode.com/problems/design-linked-list/description/?utm_source
2. Implement a program to evaluate a postfix expression.
   https://leetcode.com/problems/evaluate-reverse-polish-notation/description/
3. Queue and its operations using arrays and Linked List
   https://leetcode.com/problems/implement-queue-using-stacks/description/

   https://leetcode.com/problems/design-linked-list/description/?utm_source
4. Implement Circular Queue using Arrays
   https://leetcode.com/problems/design-circular-queue/description/

UNIT – IV

Trees: Definition of tree, Tree Terminology, Representation of tree, Types of trees-Binary Tree, BST, BST, Binary tree traversals-Inorder,Preorder,Postorder Traversals.

Binary Search Trees – Definition, Operations: Searching, Insertion, Deletion.

Practice:

1. Implement Binary search tree (BST).
   https://leetcode.com/problems/insert-into-a-binary-search-tree/description/

   https://leetcode.com/problems/delete-node-in-a-bst/description/

2. Implement Binary search Tree (BST) Traversals.
   https://www.hackerrank.com/challenges/tree-inorder-traversal/problem

   https://www.hackerrank.com/challenges/tree-preorder-traversal/problem

   https://www.hackerrank.com/challenges/tree-postorder-traversal/problem

UNIT – V

Graphs: Definition and Terminology, Types of Graph, Representation of Graphs-Adjacency Matrix and Adjaceny list, Graph Traversals (BFS & DFS), Bi-Connected Components.

Practice:

1. Graph Traversal using Breadth First Search (BFS)
   https://www.hackerrank.com/challenges/bfsshortreach/problem

2. Graph Traversal using Depth First Search (DFS)
   https://leetcode.com/problems/number-of-islands/description/

Text Books:

1. The Complete Reference C++, Herbert Schildt, TMH, Fourth Edition, ISBN-10. 0070411832.
2. Fundamentals of Data Structures in C++,Ellis Horowitz, Sartaj Sahni, DineshMehta, ISBN: 81-7515-278-8
3. Data Structures, KV Sambasivarao, S Rama Sree, S.Chand. ISBN: 978-9358704730

Reference Books:

1. The C++ Programming Language, Bjarne Stroustrup, Pearson, Fourth Edition,ISBN · 9780133522884.
2. Object oriented programming in C++, JoyceeFarrell,Cengage, ISBN : 978-1418836269
3. Data structures and algorithms in C++, 3rd edition, Adam Drozdek,Thomson, ISBN: 9788131521267
4. The Algorithm Design Manual, Steven S. Skiena, Springer Publication, Second Edition. ISBN: 978-1849967204
5. Data Structures and Algorithms by Maganti Venkatesh, Naresh. ISBN: 9780929306407

Web Links:

1. http://nptel.ac.in/courses/106105151/
2. https://nptel.ac.in/courses/106102064
3. https://archive.nptel.ac.in/courses/106/105/106105225/
4. https://www.udemy.com/topic/data-structures/
5. https://www.coursera.org/specializations/data-structures-algorithms
6. https://ds1-iiith.vlabs.ac.in/List%20of%20experiments.html

Additional Practice:

SNO	Difficulty	Problem Name	URL
1	Easy	Arrays and Hashtables	https://leetcode.com/problems/two-sum/description/
2	Easy	Arrays and Pointers	https://leetcode.com/problems/remove-duplicates-from-sorted-array/description/?utm_source
3	Medium	Arrays and Pointers	https://leetcode.com/problems/two-sum-ii-input-array-is-sorted/description/?utm_source
4	Easy	Linked List	https://leetcode.com/problems/merge-two-sorted-lists/description/
5	Medium	Arrays and Matrix	https://leetcode.com/problems/diagonal-traverse/description/
6	Easy	Linked List	https://www.hackerrank.com/challenges/compare-two-linked-lists/problem
7	Medium	Stack applications	https://www.geeksforgeeks.org/problems/tower-of-hanoi-1587115621/1?utm
8	Medium	Binary Tree	https://leetcode.com/problems/binary-tree-right-side-view/description/
9	Medium	Stack applications	https://leetcode.com/problems/remove-k-digits/description/
10	Medium	Graph Theory	https://www.hackerrank.com/challenges/journey-to-the-moon/problem
11	Medium	Graph Traversals	https://leetcode.com/problems/number-of-provinces/description/
12	Medium	Graph Theory	https://www.hackerrank.com/challenges/torque-and-development/problem

	Basic Programs	Code chef	Hacker rank	Leet code	Geek for Geeks
Unit-1	**3**	**0**	**2**	**1**	**0**
Unit-2	**0**	**1**	**3**	**2**	**0**
Unit-3	**0**	**0**	**1**	**5**	**0**
Unit-4	**0**	**0**	**3**	**2**	**0**
Unit-5	**0**	**0**	**1**	**1**	**0**
Additional Practice	**0**	**0**	**3**	**8**	**1**
Total	**3**	**1**	**13**	**19**	**1**

Business Intelligence Lab

(Common to CE, EEE, ME, ECE, CSE, IT, AIML, CSE(DS), PT & Min.E)

(Syllabus for the batch admitted in the year 2025)

	L	T	P	C
Course Code: 2501IT01	**0**	**0**	**2**	**2**

Course Outcomes:

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

CO1: Utilize Excel and Power BI for data analysis, visualization, and reporting.

CO2: Apply various data analysis techniques in Excel and Power BI to extract meaningful insights from datasets

CO3: Create clear and compelling visualizations using Excel and Power BI to communicate data-driven insights.

CO4: Develop data models in Power BI to organize and analyze data efficiently.

CO5: Design interactive dashboards in Power BI to facilitate data exploration and decision-making.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	2	2	-	-	2	-	-	-	1	1	1
CO2	2	3	-	-	2	-	-	-	1	1	2
CO3	2	2	-	-	2	-	-	-	1	1	2
CO4	1	2	1	-	2	-	-	-	1	1	2
CO5	1	2	1	-	2	-	-	-	1	1	2

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1	1	2
CO2	2	1
CO3	2	2
CO4	1	2
CO5	2	2

Practice:

1. **Introduction to Excel**
   a. Overview of Excel and its capabilities for data analysis
   b. Basics of Excel: Navigating the interface, entering data, formatting cells
   c. Introduction to functions and formulas: SUM, AVERAGE, IF, VLOOKUP, etc.
   d. d. Importing data into Excel from different sources: CSV, text files.

2. **Data Analysis with Excel**
   a. Data manipulation techniques: Sorting, filtering, and grouping data
   b. Advanced functions and formulas: INDEX/MATCH, SUMIFS, COUNTIFS etc.
   c. Data visualization with Excel: Creating charts and graphs
   d. Using PivotTables for data summarization and analysis

3. **Advanced Excel Features**
   a. Introduction to Excel tables and structured references
   b. Working with named ranges and dynamic ranges
   c. Excel data validation techniques

4. **Introduction to Power BI**
   a. Overview of Power BI and its advantages over Excel for large datasets
   b. Installing Power BI Desktop
   c. Understanding the Power BI interface: Navigation, ribbons, and panes
   d. Importing data into Power BI Desktop from various sources

5. **Data Preparation in Power BI**
   a. Introduction to Power Query for data transformation
   b. Cleaning, shaping, and filtering data in Power Query Editor

6. **Data Preparation in Power BI**
   a. Combining data from different sources
   b. Loading data into Power BI model.

7. **Data Modelling in Power BI**
   a. Understanding relationships between tables
   b. Creating calculated columns and measures using DAX

8. **Data Modelling in Power BI**
   a. Introduction to DAX functions: CALCULATE, FILTER, RELATED, etc.
   b. Working with date and time functions in DAX

9. **Visualization Basics in Power BI**
   a. Creating basic visualizations: Bar charts, line charts, pie charts, etc.
   b. Customizing visualizations: Formatting, titles, legends, etc.

10. **Visualization Basics in Power BI**
    a. Using slicers and filters to interact with visualizations
    b. Adding drilldown capabilities to visualizations

11. **Advanced Visualizations and Dashboards in Power BI**
    a. Exploring advanced visualizations: TreeMap, Waterfall chart, KPIs, etc.
    b. Creating custom visuals from the marketplace

12. **Advanced Visualizations and Dashboards in Power BI**
    a. Designing effective dashboards: Layout, arrangement, and organization
    b. Adding interactivity with bookmarks and drillthrough

Additional Practice:

1. **Basic Data Analysis:** Import a dataset into Excel and perform basic data analysis tasks such as sorting, filtering, and creating simple charts to visualize the data.
2. **Expense Tracker:** Create a spreadsheet to track your expenses. You can have columns for date, item description, category, and amount. Use formulas to calculate totals and analyze your spending habits.

3. **Data modeling and extracting statistics from dataset:** Connecting Power BI to local data files and cloud servers (COVID19 dataset will be imported into the Power BI for visualization).

Text Books:

1 Learn Power BI - Second Edition: A comprehensive, step-by-step guide for beginners to learn real-world business intelligence 2nd Edition, ISBN: 9781801811958.
2 Power BI Beginner: Zero to Hero in Power BI Desktop by Philip Seamark, ISBN: 1691641227.

Reference Books:

1 Power BI Quick Start Guide: Build dashboards and visualizations to make your data come to life" by Devin Knight and Siddharth Mehta.
2 Learn Power BI: A Beginner's Guide to Analyzing Data and Creating Reports with Power BI" by Murilo Miranda.

Web Links:

1 https://learn.microsoft.com/en-us/power-bi/
2 https://support.microsoft.com/en-us/excel
3 https://cce.sydney.edu.au/course/MSE1
4 https://cce.sydney.edu.au/course/PBBA

Business Intelligence & AI Tools Lab

(Common to CE, EEE, ME, ECE, CSE, AIML, CSE(DS), PT & Min.E)

(Revised Syllabus for batches admitted from 2026 onwards)

Course Code: 2601AI01	**L**	**T**	**P**	**C**
	0	**0**	**2**	**2**

Course Outcomes

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

CO1	Use Excel for data entry, formulas and built-in functions to perform basic data analysis.
CO2	Apply data-analysis and visualization techniques in Excel (PivotTables, charts) to extract meaningful insights from datasets.
CO3	Build data models and interactive dashboards in Power BI to support reporting and decision-making.
CO4	Use generative AI tools (ChatGPT, Claude, Gemini, Perplexity) through well-structured prompts to solve everyday academic and analytical tasks.
CO5	Apply AI tools responsibly to research, verify, summarize and create content and presentations.
CO6	Integrate AI assistance with Excel and Power BI to accelerate data analysis and reporting.

MODULE A — Spreadsheet Analytics with Microsoft Excel

1. Excel Fundamentals

a. Navigating the Excel interface; entering and formatting data; cell referencing (relative, absolute, mixed).
b. Core functions and formulas: SUM, AVERAGE, COUNT, IF, VLOOKUP.
c. Importing data into Excel from CSV and text files.

2. Data Analysis in Excel

a. Data manipulation: sorting, filtering and grouping data.
b. Advanced functions: INDEX/MATCH, SUMIFS, COUNTIFS, nested IF.
c. Creating charts and graphs to visualize data.

3. PivotTables and Summarization

a. Building PivotTables and PivotCharts for data summarization.
b. Using slicers and timelines to interact with summaries.
c. Calculated fields and grouping within PivotTables.

4. Advanced Excel Features

a. Excel tables and structured references; named and dynamic ranges.

b. Data-validation techniques (drop-down lists, input rules).
c. What-if analysis: Goal Seek and Data Tables.

MODULE B — Business Intelligence with Microsoft Power BI

5. Getting Started with Power BI

a. Overview of Power BI and its advantages over Excel for large datasets.
b. Installing Power BI Desktop; understanding the interface (ribbons, panes, views).
c. Importing data from multiple sources (Excel, CSV, web).

6. Data Preparation with Power Query

a. Cleaning, shaping and filtering data in the Power Query Editor.
b. Combining data from different sources (append and merge).
c. Loading the prepared data into the Power BI model.

7. Data Modeling and DAX

a. Creating relationships between tables; star-schema basics.
b. Calculated columns and measures using DAX.
c. Key DAX functions: CALCULATE, FILTER, RELATED, and date/time functions.

8. Visualizations and Interactive Dashboards

a. Basic and advanced visuals: bar, line, pie, TreeMap, KPI cards.
b. Slicers, filters and drill-down for interactivity.
c. Designing an interactive dashboard with bookmarks and drill-through.

MODULE C — Working with AI Tools

9. Exploring the AI Tool Landscape

a. Introduction to popular AI assistants: ChatGPT, Claude, Gemini, Perplexity.
b. Comparing tools across tasks — summarizing, simple coding, and reasoning.
c. Building a comparison matrix and choosing the right tool for a task.

10. Prompt Engineering Essentials

a. Anatomy of a good prompt; clarity, context and constraints.
b. Prompt patterns: persona, few-shot, chain-of-thought, and structured (JSON) output.
c. Iterating and refining prompts for better results.

11. AI for Research and Verification

a. Using AI for topic research, explanations and quick literature scans.
b. Three-step verification chain: cross-check claims against primary sources.
c. Recognizing hallucinations, bias and link-rot; the limits of AI.

12. Responsible and Ethical Use of AI

a. Academic integrity, plagiarism and proper acknowledgement of AI use.
b. Privacy, data safety and never sharing sensitive information.
c. Drafting a simple personal/class “AI use policy”.

13. AI for Content and Presentations

a. Summarizing documents and Q&A over your own notes (e.g., NotebookLM).
b. Generating outlines, write-ups and email drafts with AI assistance.
c. Auto-creating slide decks from a brief (e.g., Gamma) and refining them.

14. AI-Assisted Data Analysis (Capstone)

a. Using AI to write/explain Excel formulas and draft DAX measures.
b. Cleaning and interpreting a dataset and explaining charts with AI help.
c. Mini-project: clean a dataset in Excel, build a Power BI dashboard, and use AI to generate insights and a short summary report.

Cornerstone projects :

S. No	Project Title	Project Description	Dataset URL
1	**Indian General Election Analytics Dashboard**	Develop an interactive dashboard to analyze election results across all Indian states and union territories. The project includes Spreadsheet Analytics using Microsoft Excel, Business Intelligence using Microsoft Power BI, and AI insights to visualize ruling parties, constituencies won, vote share, voter turnout, alliance performance, and state-wise electoral trends through interactive reports and maps.	https://results.eci.gov.in/
2	**Global Renewable Energy Analytics Dashboard**	Design a Business Intelligence dashboard to analyze renewable energy production across countries. The project focuses on energy generation by source, installed capacity, consumption patterns, and sustainability trends using Excel, Power BI dashboards, and AI-driven insights for comparative analysis.	https://ourworldindata.org/energy
3	**Air Quality Index (AQI) Analytics Dashboard**	Build an interactive dashboard to monitor air pollution levels across cities and regions. Analyze AQI values, pollutant concentrations (PM2.5, PM10, NO_2, SO_2), seasonal trends, and pollution hotspots using Excel, Power BI visualizations, and AI-powered insights.	https://www.kaggle.com/datasets/rohanrao/air-quality-data-in-india
4	**FIFA World Cup Analytics Dashboard**	Create a comprehensive dashboard to analyze FIFA World Cup tournaments by exploring team performance, match results, goals scored, player statistics, venues, and championship history. The project utilizes Excel, Power BI, and AI insights to identify historical trends and performance patterns.	https://www.kaggle.com/datasets/abecklas/fifa-world-cup
5	**Global Earthquake Analytics Dashboard**	Develop a dashboard that visualizes earthquake occurrences worldwide by analyzing magnitude, depth, geographic location, frequency, and affected regions. Excel is used for data preparation, while Power BI and AI insights help identify seismic trends and risk-prone areas.	https://earthquake.usgs.gov/earthquakes/feed/v1.0/csv.php

6	**Flight Delay Analytics Dashboard**	Design an interactive airline performance dashboard to evaluate flight delays, cancellations, airline efficiency, airport performance, and seasonal travel trends. The project combines Excel, Power BI, and AI insights to support operational analysis and decision-making.	https://www.kaggle.com/datasets/usdot/flight-delays
7	**Global CO_2 Emissions Analytics Dashboard**	Analyze global carbon dioxide emissions by country, region, and industry using Power BI. The dashboard presents emission trends, sector-wise contributions, historical comparisons, and sustainability indicators through Excel, interactive visualizations, and AI-generated insights.	https://ourworldindata.org/co2-and-greenhouse-gas-emissions
8	**Olympics Performance Analytics Dashboard**	Create a Power BI dashboard to evaluate Olympic Games performance by analyzing medal counts, athlete achievements, country rankings, sports categories, and historical participation. Excel and AI-powered visualizations are used to generate meaningful sports analytics and comparative insights.	https://www.kaggle.com/datasets/the-guardian/olympic-games
9	**Netflix Content Analytics Dashboard**	Develop an interactive dashboard to analyze Netflix's content library by genre, country, release year, ratings, duration, and content trends. Excel is used for preprocessing, while Power BI and AI insights help identify viewing patterns and content distribution across regions.	https://www.kaggle.com/datasets/shivamb/netflix-shows
10	**Global Education Statistics Dashboard**	Build a Business Intelligence dashboard to analyze global education indicators such as literacy rates, school enrollment, graduation rates, education expenditure, and gender parity. Using Excel, Power BI, and AI insights, the dashboard supports comparative analysis of educational development across countries.	https://data.worldbank.org/topic/education
11	**Global Food Production Analytics Dashboard**	Develop an interactive dashboard to analyze global food production, crop yields, agricultural output, and production trends across countries. The project incorporates Spreadsheet Analytics using Microsoft Excel, Business Intelligence using Microsoft Power BI, and AI-powered insights to identify production patterns, top-producing countries, and long-term agricultural trends through interactive visualizations.	https://ourworldindata.org/food-supply
12	**Road Traffic Fatalities Analytics Dashboard**	Design a Business Intelligence dashboard to analyze road traffic accidents and fatalities across countries. The dashboard visualizes accident rates, fatalities by road users, regional comparisons, and yearly	https://www.kaggle.com/datasets/data125661/india-road-accident-dataset

		trends using Microsoft Excel, Power BI, and AI insights to support road safety analysis and policy evaluation.	
13	**Global Population Demographics Dashboard**	Build a comprehensive dashboard to analyze population growth, age distribution, urbanization, life expectancy, fertility rates, and demographic changes across countries. The project uses Excel for data preparation, Power BI for visualization, and AI insights to identify demographic trends and support population-based decision-making.	https://ourworldindata.org/population
14	**International Tourism Analytics Dashboard**	Create an interactive dashboard to evaluate global tourism trends by analyzing international tourist arrivals, tourism receipts, popular destinations, and regional tourism growth. Using Microsoft Excel, Power BI, and AI-driven insights, the dashboard enables comparative analysis of tourism performance across countries and years.	https://data.worldbank.org/indicator/ST.INT.ARVL
15	**Global Internet Usage & Digital Connectivity Dashboard**	Develop a Business Intelligence dashboard that analyzes internet penetration, broadband subscriptions, mobile connectivity, and digital adoption across countries. The project combines Spreadsheet Analytics in Microsoft Excel, interactive Power BI dashboards, and AI-generated insights to explore global digital transformation and connectivity trends.	https://data.worldbank.org/indicator/IT.NET.USER.ZS

Text Books

1 Learn Power BI – Second Edition: A comprehensive, step-by-step guide for beginners to learn real-world business intelligence, ISBN: 9781801811958.

2 Power BI Beginner: Zero to Hero in Power BI Desktop, Philip Seamark, ISBN: 1691641227.

3 Excel 2021 / Microsoft 365 – The Step-by-Step Guide to Mastering Excel, (latest edition).

Reference Books

1 Power BI Quick Start Guide: Build dashboards and visualizations to make your data come to life, Devin Knight and Siddharth Mehta.

2 Learn Power BI: A Beginner's Guide to Analyzing Data and Creating Reports with Power BI, Murilo Miranda.

3 Co-Intelligence: Living and Working with AI, Ethan Mollick (for responsible AI use).

Web Links / e-Resources

1 https://learn.microsoft.com/en-us/power-bi/

2 https://support.microsoft.com/en-us/excel

3 https://aistudio.google.com/

4 https://notebooklm.google.com/

5 https://www.perplexity.ai/

Note:

- The cornerstone project team size shall be four (4) students.
- Students may select any one of the above prescribed projects or a project of their own choice with the prior permission of the Course instructor.
- The evaluation of the Cornerstone Project shall be based on the project implementation, oral presentation, 10-minute video presentation, report and viva voce.
- The video presentation should consist of the working procedure of the project along with contribution of each student for a minimum of 2 minutes.
- Finally, the Source code of the cornerstone Project has to be pushed into the Students GitHub repository (if applicable).

Discrete Mathematics
(Common to CSE, IT, AIML & CSE (DS))

	L	T	P	C
Course Code: 2501MA08	**2**	**1**	**0**	**3**

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

CO1: Apply the principles of mathematical logic to statement calculus and predicate calculus.
CO2: Compute transitive closure, equivalence classes of binary relations.
CO3: Apply basic counting techniques to solve the combinatorial problems.
CO4: Solve recurrence relations by various methods.
CO5: Apply the concepts of Graph theory to find Euler paths, Hamiltonian paths and demonstrate different traversal methods for trees.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	3	2	-	-	-	-	-	-	-	-	-
CO2	3	2	-	-	-	-	-	-	-	-	-
CO3	2	3	-	-	-	-	-	-	-	-	-
CO4	3	2	-	-	-	-	-	-	-	-	-
CO5	3	2	-	-	-	-	-	-	-	-	-

UNIT-I

Logic: Propositional Logic: Statements and logical connectives, Well-Formed Formulas, Truth Tables, Tautologies, Equivalence of Formulas, Duality Law, Tautological Implications, Normal Forms, Theory of Inference for Statement Calculus. Predicate Logic: Statement Functions, Variables, and Quantifiers; Inference Theory for Predicate Calculus.

Practice(Using any computational tool) :

1. Construct the truth table calculator taking two inputs, P and Q, and calculate AND, OR and NOT.

UNIT-II

Relations: Properties of Binary Relations, Relation Matrix, Digraph, Operations on Relations, Partition and Covering, Transitive, Closure, Warshall Algorithm, Equivalence Relation, R-Equivalence Class, partially ordered sets, Hasse Diagrams.

UNIT-III

Combinatorics: Basis of Counting, Permutations, Permutations with Repetitions, Circular and Restricted Permutations, Combinations, Restricted Combinations, Binomial and Multinomial Coefficients and Theorems.

UNIT-IV

Recurrence Relations: Formation of Recurrence Relations, Solving Recurrence Relations by Substitution and Generating Functions, Method of Characteristic Roots.

Practice (Using any computational tool):

1. Compute the n^{th} Fibonacci number

UNIT – V

Graph Theory: Basic Concepts of Graphs, Matrix Representation of Graphs: Adjacency Matrix, Incidence Matrix, Isomorphic Graphs, Paths and Circuits, Euler and Hamilton Graphs, Planar Graphs, and Euler's Formula.
Trees: Tree Properties, Spanning Trees, BFS Algorithm, DFS Algorithm, Minimal Spanning Trees, and Kruskal's Algorithm, Graph coloring, chromatic number.

Practice (Using any computational tool):

1. BFS and DFS algorithms.

Text Books:

1 Discrete Mathematics and its Applications with Combinatorics and Graph Theory, Kenneth H. Rosen, McGraw Hill Education (India) Private Limited.ISBN 9780070681880**.**
2 Discrete Mathematical Structures with Applications to Computer Science, J. P. Tremblay and R. Manohar, Tata Mc Graw Hill.ISBN 978-0074631133

Reference Books:

1 Discrete Mathematical Structures, Bernand Kolman, Robert C. Busby and Sharon Cutler Ross, PHI. ISBN 978-9332549593.
2 Mathematical Foundations of Computer Science, S. Santha, E. V. Prasad, Cengage Publishers. ISBN 978-8131515938.

Web Links:

1 http://nptel.ac.in/courses/106106094/
2 http://mathworld.wolfram.com/classroom/classes/DiscreteMathematics.html
3 http://mathworld.wolfram.com/topics/GeneralLogic.html
4 https://ggc-discrete-math.github.io/

Probability & Statistics
(Common to CSE, IT, AIML & CSE(DS))

	L	T	P	C
Course Code: 2501MA09	**2**	**1**	**0**	**3**

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

CO1: Apply Baye's theorem
CO2: Apply various Probability distributions for both discrete and continuous random variables
CO3: Compute mean and variance of sample means with replacement and without replacement and maximum errors.
CO4: Apply various tests to test the hypothesis concerning mean, Proportion,variance and attributes.
CO5: Apply the concepts of correlation and regression to the given statistical data.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	2	3	-	-	-	-	-	-	-	-	-
CO2	3	2	-	-	-	-	-	-	-	-	-
CO3	2	3	-	-	-	-	-	-	-	-	-
CO4	3	2	-	-	-	-	-	-	-	-	-
CO5	3	2	-	-	-	-	-	-	-	-	-

UNIT-I

Probability and Random Variables:
Probability – Conditional probability and Baye's Theorem-Random variables – Discrete and Continuous random variables-Mathematical Expectation, Variance and Moment generating function.

Practice(Using any computational tool):
1. Calculation of Mean and Variance.

UNIT-II

Probability Distributions:
Distribution function –Binomial, Poisson, Normal and Exponential distributions.

Practice(Using any computational tool):
1. Discrete and continuous probability distributions (Binomial, Poisson, normal and exponential distributions)

UNIT-III

Data Analytics and Sampling Theory:
Data science –Collection of data – primary and secondary data – Types of variables:

dependent and independent Categorical and Continuous variables – Data visualization – Measures of Central tendency(Mean,mode and median) – Measures of Variability (spread or variance).
Sampling Theory: Introduction – Population and samples – Sampling distribution of Means and Variance – Central limit theorem (without proof)-Point and Interval estimations – Maximum error of estimate.

Practice (Using any computational tool):

1. Finding measures of variability and central tendency (mean, median, mode, arithmetic mean, geometric mean, Variance)
2. Data visualization (Construction of bar charts, pie charts).
3. Construction of confidence intervals

UNIT-IV

Tests of Hypothesis: Introduction – Hypothesis – Null and Alternative Hypothesis – Type I and Type II errors – Level of significance – One tail and two-tail tests – Tests concerning one mean and two means (Large and Small samples) – Tests on proportions, Chi-Square and F test.

Practice(Using any computational tool):

1. Z-test, t-test and Chi-square test

UNIT – V

Correlation and Regression: Method of least squares – Straight line - nonlinear curves– parabola -Exponential – Power curves-Correlation – Karl Pearson's correlation coefficient – rank correlation – regression-– regression coefficients and properties (without proof) – regression lines.

Practice(Using any computational tool):

1. Finding correlation coefficient and regression lines.

*Students are advised to use any Programming language/computational tool/AI tool like Python,Excel, PowerBI ,Wolfrum Alpha, Symbolab, Mathway, etc., for the practice

Text Books:

1 Miller and Freund's, Probability and Statistics for Engineers, 9/e, Pearson, 2020. ISBN 978-9353945237
2 Fundamentals of Mathematical Statistics, S.C. Gupta and V.K.Kapoor, 12th edition, Sultan Chand & Sons Publications, 2020.ISBN **978-9351611738**

Reference Books:

1 Probability, Statistics and Random processes, T.Veerrajan, Tata McGraw-Hill.ISBN 978-0070669253
2 Probability and statistics by T.K.V. Iyengar, S. Chand publishers,2022 ISBN 9789355010643
3 Modern Statistics: Intuition, Math, Python, R by Mike X Cohen.ISBN 9798867723736

Web Links:

1. https://onlinecourses.nptel.ac.in/noc21_ma74/preview
2. http://mathworld.wolfram.com/topics/ProbabilityandStatistics.html
3. https://www.khanacademy.org/math/statistics-probability
4. https://www.geeksforgeeks.org/statistics-with-python/
5. https://www.coursera.org/learn/inferential-statistical-analysis-python?specialization=statistics-with-_python

Artificial Intelligence
(Common to CSE, IT, AIML & CSE(DS))

	L	T	P	C
Course Code: 2501AI02	**2**	**0**	**1**	**3**

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

CO1: Illustrate the foundations, history, and applications of Artificial Intelligence using modern tools
CO2: Develop AI-based problem-solving models using Python
CO3: Implement search and optimization algorithms using Python libraries
CO4: Apply knowledge representation techniques using Python frameworks
CO5: Design simple intelligent systems and expert systems using Python

Mapping of Course Outcomes with Program Outcomes:

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

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1	1	
CO2	2	1
CO3	1	1
CO4	1	
CO5	1	1

Unit - I

Introduction to AI & Python Foundations

What is AI? Foundations and History of AI, AI Applications in real-world systems, Intelligent Agents and Environment, Rationality and Types of Agents, Fundamentals of Python for AI applications.

Practice (Python-based):

1. Implement simple AI agent (e.g., vacuum cleaner agent simulation)
2. Rule-Based AI Agent (Weather Decision Agent)

Unit – II

Problem Solving & Search Algorithms

Problem formulation and state space representation, Control strategies, Search strategies, Uninformed Search: BFS, DFS, Uniform Cost Search, Informed Search: Greedy, A* Algorithm, Heuristics and evaluation functions.

Practice (Python):

1. Implement DFS algorithm for Water Jug Problem
2. Solve Tic-Tac-Toe using BFS search strategy.
3. Implement A* Algorithm for pathfinding (e.g., grid navigation)

Unit – III

Optimization & Game Playing

Problem graphs, Matching, Indexing, Local Search Algorithms: Hill Climbing, Simulated Annealing, Optimization Problems, Searching with Nondeterministic Actions, Constraint Satisfaction Problems (CSP), Game Playing: Minimax Algorithm, Alpha-Beta Pruning

Practice (Python):

1. Solve 8-Puzzle using Hill Climbing
2. Implement Travelling Salesman Problem (TSP) using heuristic approach

Unit – IV

Knowledge Representation

Knowledge Representation Techniques, Propositional Logic & Predicate Logic, Inference & Resolution, Semantic Networks and Frames, Representation of knowledge through Planning generation Systems-K Strips Strategic explanations Why, why not and how explanations.

Practice (Python):

1. a) Implement Predicate Logic using Python (SymPy / logic module)
   b) Develop a Semantic Network (e.g., Animals or Transport)
2. Create a Frame-based system (Restaurant domain)

Unit – V

Expert Systems & Modern AI Applications

Expert Systems Architecture, Knowledge Acquisition & Representation, Meta knowledge, Rule-based Systems, Ethics in AI, Typical expert systems – EMYCIN, DART, XOON, Expert systems shells.

Practice (Python):

1. Build a Travel Recommendation Expert System
2. Develop a Student Career Guidance System

Text Books

1. Artificial Intelligence: A Modern Approach – Stuart Russell & Peter Norvig (need to add details)
2. Artificial Intelligence: Foundations of Computational Agents – Poole & Mackworth

Reference Books

1. Artificial Intelligence with Python – Prateek Joshi
2. Hands-On Machine Learning with Scikit-Learn & TensorFlow – Aurélien Géron
3. Python AI Projects for Beginners – Prateek Joshi

Web Resources

1. https://docs.python.org
2. https://nptel.ac.in/courses/106105077
3. https://nptel.ac.in/courses/106106126

Object Oriented Programming through C++
(Common to CSE, IT, AIML & CSE(DS))

	L	T	P	C
Course Code: 2501CS08	**2**	**0**	**2**	**4**

Course Outcomes:

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

CO1: Make use of C and C++ programming constructs to demonstrate OOP concepts.
CO2: Develop applications using constructor and destructor.
CO3: Apply C++ features for problem solving.
CO4: Apply inheritance concepts and file I/O to solve a given problem.
CO5: Design C++ classes using templates and STL.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	2	1	2	-	-	-	-	1	1	-	-
CO2	1	2	2	-	-	-	-	1	1	-	-
CO3	1	1	1	-	-	-	-	1	1	-	-
CO4	1	1	2	-	-	-	-	1	1	-	-
CO5	2	1	2	-	-	-	-	1	1	-	-

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1	2	-
CO2	2	-
CO3	2	-
CO4	2	-
CO5	2	-

UNIT – I

Introduction to C++: Differences between C and C++, Disadvantages of Conventional Programming, Evolution of C++, Key Concepts of Object Oriented Programming, Structure of C++ program , Advantages of OOP, scope resolution Operator, Namespace.
Practice:

7. Find the roots of a quadratic equation.
8. Find factorial of a given number using recursion
9. Implement scope resolution and namespaces.
10. Illustrate the use of default arguments and access specifiers

UNIT – II

Classes and Objects: Declaring Objects- Access Specifiers and their Scope,Defining Member Function, Rules for Inline Functions, static Member Variable, static Member Function, Friend functions, overloading Member Function, Array of Objects, Object as Parameter, Object as return type, Nestedclass, Introduction to Constructors and Destructor,Characteristics of Constructor and Destructor, Application with Constructor, Constructor with Arguments, Copy Constructor, Anonymous Objects.

Practice:

1. Program to illustrate inline functions and function overloading.
2. Program to illustrate friend function
3. Program to illustrate the use of Constructors and Destructors.
4. Program illustrating Constructor overloading.
5. Program illustrating Copy Constructor.

UNIT – III

Operator Overloading: Introduction, The Keyword Operator,Rules for Overloading Operators,Overloading Unary Operator,Overloading Binary Operator, Overloading using friend function.
Inheritance: Introduction, Reusability, Types of Inheritance, Virtual Base Classes, Object as a Class Member, Abstract Classes, Advantages and Disadvantages of Inheritance. Pointer to Class- Pointer Object, this Pointer, Pointer to Derived Classes and Base Class.
Binding Polymorphisms and Virtual Functions: Introduction, Binding in C++,Virtual Functions, Rules for Virtual Function, Abstract classes, Virtual Destructor

Practice:

1. Overload Unary, and Binary Operators using member function.
2. Overload Unary, and Binary Operators using friend function.
3. Incorporate various forms of Inheritance i. Single Inheritance ii. Multiple Inheritances
   iii. Multi-level inheritance iv. Hierarchical Inheritance v. Hybrid inheritance
4. Order of execution of constructors and destructors in inheritance.
5. Illustrate the use of object as a class member,pointer to a class,this pointer and Virtual Base Class.
6. Illustrate virtual functions.
7. Implement pure virtual function and calculate the area of different shapes by using abstract class.

UNIT – IV

Generic Programming with Templates: Introduction, Need for Templates,Definition of class Templates, Normal Function Templates, Overloading of Template Function, Bubble Sort Using Function Templates, Difference between Templates and Macros, Linked list with

templates

Practice:

1. Illustrate the use of function template.
2. Implement template class.
3. Implement class templates with multiple parameters.

UNIT – V

Exception Handling: Introduction, Principles of Exception Handling, The Keywords try throw and catch, Multiple Catch Statements.
Files: Stream I/O ,Reading and writing to Files.
Overview of Standard Template Library: Introduction, STL Programming Model, Containers, Sequence Containers, Associative Containers, Algorithms, Iterators, Vectors, Lists, Maps

Practice:

1. Implement Exceptions handling.
2. Illustrate the use of multiple catch statements.
3. Implement List,Vector and its Operations.
4. Implement Deque and its Operations.
5. Implement Map and its Operations.

Additional Practice:

1. Develop a C++ program for flight booking system
2. Develop Qt application containing slider and spin box in which a slider responds to changes in the spin box.
3. Develop a Qt application for creating a text pad.
4. Develop a C++ program with maximum of 20 characters, that your user will be guessed and will show
   only asterisks (*) on the screen. The user will input or enter one character at a time. And for every correct character, the asterisk will be replaced by that character until all the characters or the mystery word/s will reveal. Your program will accept a maximum three (3) errors or mistakes in entering/inputting character otherwise the mystery word/s will be viewed. Sample Output: Output: ******* Enter your character: e
   Output: ***e**e
   Enter your character: a Output: sorry! the character is not existing. you still have 2 chances
   Enter your character: s Output: s**e**e
   Enter your character: c Output: sc*e*ce
   Enter your character: i Output: scie*ce
   Enter your character: n Output: science

Text Books:

1. The Complete Reference C++, Herbert Schildt, Fourth Edition, TMH ISBN-13:978-0070532465, and the ISBN-10 is 007053246X.
2. The C++ Programming Language, Bjarne Stroustrup, Fourth Edition, Pearson ISBN 978-0321563842

Reference Books:

1. A First Book Of C++, Gary Bronson, Fourth Edition, Cengage Learning ISBN-10. 8131532712 and ISBN-13. 978-8131532713
2. C++ Primer Plus By Stephen Prata, Sixth Edition, Pearson ISBN: 978-0321776402
3. C++ Programming:From Problem Analysis To Program Design, Ds Malik,Eighth Edition, Cengage Learning ISBN: 9781337102087

Web Links:

1. http://nptel.ac.in/courses/106105151/
2. http://www.geeksforgeeks.org/c-plus-plus/
3. https://infyspringboard.onwingspan.com/web/en/viewer/html/lex_auth_013501572848615424101 01
4. https://www.udemy.com/course/beginning-c-plus-plus-programming/

Database Management Systems
(Common to CSE, IT, AIML & CSE(DS))

	L	T	P	C
Course Code: 2501IT05	**2**	**0**	**2**	**4**

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

CO1: Describe the Fundamental concepts of DBMS.
CO2: Interpret relational database using SQL.
CO3: Make use of normalization techniques for database design.
CO4: Illustrate the mechanisms of transaction management.
CO5: Optimize database performance with advanced indexing, query optimization, and robust backup and recovery strategies.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	3	3	3	-	1	-	-	1	-	-	-
CO2	3	3	3	2	2	1	-	1	-	-	
CO3	3	3	3	3	3	2	-	-	-	-	1
CO4	3	3	3	1	2	1	-	-	-	-	-
CO5	3	3	3	3	3	2	-	-	-	-	2

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1	3	3
CO2	3	2
CO3	3	2
CO4	3	1
CO5	3	2

UNIT – I
Introduction to DBMS: History and Architecture,Need and Purpose of DBMS, Applications of DBMS, Data and Metadata, Levels of abstraction,Three-Schema Architecture, structure of DBMS, Database Users & Roles of DBA, Basics of Parallel and Distributed Databases, Specialty Databases – Spatial, Temporal, Multimedia, NoSQL.
View of Data : Concepts of Schema, Data Models(Relational, ER, Object-based, Semi-structured), Steps in Database Design, Database system environment, Centralized and Client Server architecture for the database**.**

Practice

1. Familiarization with installation of DBMS(oracle 10g).
2. Draw the schema model for a University Database System by identifying the main entities using draw.io/diagrams.net or any similar tool.

UNIT – II

Introduction to the Relational Model: Structure of Relational Model**,** concepts of domain, attribute, tuple, relation, Database Schema and Schema Diagrams, Keys in Relational Model: Primary, Candidate, Alternate, Foreign, Importance and Handling of NULL Values, Relational Algebra: Fundamental Operations.

Basic SQL: Data types, table definitions (create, alter), DML operations (insert, delete, update), basic SQL querying (select, where clause, projection), arithmetic&logical operations, SQL functions, Introduction to GROUP BY and HAVING clauses.

Practice

1. Querying and modifying the database using Data Manipulation Language commands - select, insert, update, delete.
2. Implementation of SQL Functions –Use group-by and having clause.

- Find Customer Referee using DML Operations
    - https://leetcode.com/problems/find-customer-referee/description/?envType=study-plan-v2&envId=top-sql-50
- Average Time of Process per Machine using aggregate functions.
    - https://leetcode.com/problems/average-time-of-process-per-machine/description/?envType=study-plan-v2&envId=top-sql-50
- DML Operations on given schema
    - https://www.hackerrank.com/challenges/revising-the-select-query-2/problem?isFullScreen=true
- Aggregate Operators
    - https://www.hackerrank.com/challenges/earnings-of-
    - employees/problem?isFullScreen=true
- Basic Queries for salaries of employees
    - https://www.hackerrank.com/challenges
- /salary-of-employees/problem?isFullScreen=true

UNIT – III

ER Model – Introduction to entities, attributes, entity sets, relationships, relationship sets, constraints. Specialization, generalization, inheritance, subclasses and super classes using

ER diagrams.

Advanced SQL – Creating tables with relationships, implementing key and integrity constraints, nested and correlated subqueries, grouping, aggregation, ordering, different types of joins (natural, equi, outer, left, right, inner), views (updatable and non-updatable), relational set operations.

Practice:

1. Perform join operations: natural, equi-join, outer join, left, right, inner; assess query performance.
2. Perform set operations: union, intersection, difference.
3. Implement nested and correlated subqueries.
4. Creating and querying views and materialized views.

- Queries using join operations Ex: Product Sales Analysis I
    - https://leetcode.com/problems/product-sales-analysis-i/description/?envType=study-plan-v2&envId=top-sql-50
- Queries using Nested queries. Ex: Employees Whose Manager Left the Company
    - https://leetcode.com/problems/employees-whose-manager-left-the-company/description/?envType=study-plan-v2&envId=top-sql-50
- Queries using join operations
    - https://leetcode.com/problems/primary-department-for-each-employee/description/?envType=study-plan-v2&envId=top-sql-50
- Queries challenge the pads problem
    - https://www.hackerrank.com/challenges/ the-pads/problem?isFullScreen=true
- Queries type of triangle problem
    - https://www.hackerrank.com/challenges/ what-type-of-triangle/problem?isFullScreen=true
- Joins
    - https://www.hackerrank.com/challenges/the-report/problem?isFullScreen=true

UNIT – IV

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

Transaction Concept: Transaction State, Implementation of Atomicity and Durability, Concurrent Executions, Serializability, Recoverability, Storage, Recovery and Atomicity, Recovery algorithm.

Practice

1. **Implement SQL queries on a normalized database schema** using the given university database structure. The schema consists of the following tables:

- **Students** (StudentID, StudentName, Major)
- **Courses** (CourseID, CourseName, Credits)

- **Enrollments** (StudentID, CourseID, EnrollmentDate)
- **Instructors** (InstructorID, InstructorName, Phone)
- **Course_Instructors** (CourseID, InstructorID)

2. (a) Implementation of Data Control Language commands – grant and revoke.
(b) Implementation of Transaction Control Language commands - commit, savepoint, and rollback.
- Understanding Transaction, Commit and Rollback
    - https://www.codechef.com/learn/course/sql/SQ00LS10A/problems/TRA01
- Queries on normalized database.
    - https://www.hackerrank.com/challenges/symmetric-pairs/problem?isFullScreen=true
- Queries on occupations
    - https://www.hackerrank.com/challenges/occupations/problem?isFullScreen=true

UNIT – V

Indexing Techniques: Introduction**,** B+ Trees: Search, Insert, delete algorithms, File Organization and Indexing, Cluster Indexes, Primary and Secondary Indexes, Index data Structures, Hash Based Indexing: Tree base Indexing, Comparison of File Organizations.

Database Tuning: Introduction, Database Performance Tuning, Query Optimization: Introduction, Query Optimization algorithms, Backup and Recovery.

Practice

1. Create a Primary, Secondary Index on a Column.
2. Retrieve Data Using an Index.
3. Insert Data and Update Indexes.
4. Delete Data and Impact on Indexes.

*** Note:** The student must Complete & Submit a Database Foundations Certificate Course offered by Oracle Academy at the end of the Practice Session.

Capstone Project:

Project 1: University Academic Management

Project 2: Hospital Management System

Project 3: E-Commerce Order Management System

Project 4: Library Management System

Project 5: Bank Account Management System

Project 6: Online Shopping System

Project 7: Software Management System

Students should maintain for the project:

Data (Database)

- Identify at least 6 entities for your domain (e.g., Student, Course, Faculty, Department, Enrollment, Results)

- Define all attributes for each entity – simple, composite, multi-valued, and derived attributes
- Maintain tables e.g., Students, Courses, Faculty, Enrollment, and Results
- Ensure proper relationships (Student–Course via Enrollment, Faculty–Course)
- Store accurate data with primary keys, foreign keys, and constraints
- Avoid redundancy through normalization (up to 3NF)
- Maintain data integrity and validation (no duplicate IDs, valid grades)

Documentation

- Clear objective and problem definition
- Well-defined database schema (tables & attributes)
- ER diagram showing entities and relationships
- Explanation of modules and functionalities
- Description of SQL operations and queries used
- Sample outputs/screenshots
- Mention of enhancements and improvements

Textbooks:

1 Database System Concepts ,Abraham Silberschatz, Henry Korth, and S. Sudarshan, McGraw-Hill , 8th Edition, ISBN-13**:** 978-1260230508.

2 Database Management Systems, Raghurama Krishnan, Johannes Gehrke, Mc Graw-Hill, 4th Edition, ISBN: 978-1260091195.

Reference Books:

1 Database Systems, Ramez Elmasri, Shamkant B. Navathe, 7th Edition, Pearson, ISBN: 978-0137504277.

2 Database Systems, Carlos Coronel, Steven Morris, Peter Rob, 10th Edition, Cengage, ISBN: 978-0357517864.

3 Introduction to Database Systems, C J Date, Pearson, 9th Edition, ISBN: 978-0133970777.

Web Links:

1 https://academy.oracle.com/pages/coursedescription/Oracle%20Academy%20Database%20Foundations%20Course%20Description.pdf
2 https://www.w3schools.com/sql/
3 https://onlinecourses.nptel.ac.in/noc22_cs91/

Additional Practice:

S.No	Difficulty	Problem Name	Link
1.	Easy	Select	https://www.codechef.com/learn/course/sql/SQ00LS04/problems/GSQ17
2.	Easy	DML queries	https://leetcode.com/problems/second-highest-salary/description/
3.	Easy	Select All	https://www.hackerrank.com/challenges/select-all-sql/problem?isFullScreen=true
4.	Easy	Basic employee names	https://www.hackerrank.com/challenges/name-of-employees/problem?isFullScreen=true
5.	Easy	Aggregate functions	https://www.hackerrank.com/challenges/revising-aggregations-sum/problem?isFullScreen=true
6.	Easy	Basic SQL Querying	https://www.codechef.com/learn/course/sql/SQ00LS04/problems/GSQ21
7.	Easy	Basic Join	https://www.hackerrank.com/challenges/the-report/problem?isFullScreen=true
8.	Easy	Basic Join	https://www.hackerrank.com/challenges/average-population-of-each-continent/problem?isFullScreen=true
9.	Easy	Views	https://leetcode.com/problems/article-views-i/
10.	Easy	Database Normalization	https://www.hackerrank.com/challenges/placements/problem?isFullScreen=true
11.	Easy	1NF	https://www.hackerrank.com/challenges/database-normalization-1-1nf/problem?isFullScreen=true
12.	Easy	1NF/2NF/3NF	https://www.hackerrank.com/challenges/database-normalization-123nf/problem?isFullScreen=true
13.	Easy	2NF/3NF	https://www.hackerrank.com/challenges/database-normalization-4/problem?isFullScreen=true
14.	Easy	Select	https://www.hackerrank.com/challenges/average-population-of-each-continent/problem?isFullScreen=true
15.	Easy	Select ID	https://www.hackerrank.com/challenges/select-by-id/problem?isFullScreen=true
16.	Easy	attibutes	https://www.hackerrank.com/challenges/japanese-cities-attributes/problem?isFullScreen=true
17.	Easy	Select	https://www.hackerrank.com/challenges/select-by-id/problem?isFullScreen=true

18.	Easy	Select	https://www.hackerrank.com/challenges/weather-observation-station-3/problem?isFullScreen=true
19.	Easy	Select	https://www.hackerrank.com/challenges/weather-observation-station-1/problem?isFullScreen=true
20.	Easy	Select	https://www.hackerrank.com/challenges/more-than-75-marks/problem?isFullScreen=true
21.	Easy	Indexing	https://leetcode.com/problems/create-a-new-column/description/
22.	Medium	BCNF	https://www.hackerrank.com/challenges/database-normalization-5/problem?isFullScreen=true
23.	Medium	4NF	https://www.hackerrank.com/challenges/database-normalization-10/problem?isFullScreen=true
24.	Medium	select	https://www.hackerrank.com/challenges/the-company/problem?isFullScreen=true
25.	Medium	joins	https://www.hackerrank.com/challenges/harry-potter-and-wands/problem?isFullScreen=true
26.	medium	Indexing	https://leetcode.com/problems/primary-department-for-each-employee/description/
27.	Medium	Group BY	https://leetcode.com/problems/product-price-at-a-given-date/description/
28.	Medium	Group BY	https://leetcode.com/problems/group-sold-products-by-the-date/description/
29.	Hard	Delete index	https://leetcode.com/problems/delete-duplicate-emails/description/
30.	Hard	Insert and update	https://leetcode.com/problems/trips-and-users/description/
31.	Hard	Insert and update	https://leetcode.com/problems/students-and-examinations/description/
32.	Hard	Delete index	https://leetcode.com/problems/immediate-food-delivery-ii/description/
33.	Hard	Views	https://www.hackerrank.com/challenges/interviews/problem?isFullScreen=true
34.	Hard	joins	https://www.hackerrank.com/challenges/15-days-of-learning-sql/problem?isFullScreen=true

	Aditya's Internal Coding Platform (Maya)	Code chef	Hacker rank	Leet code	Total
Unit-1	0	0	0	0	**0**
Unit-2	0	0	3	2	**5**
Unit-3	0	0	3	3	**6**
Unit-4	0	1	2	0	**0**
Unit-5	0	0	0	0	**0**
Additional Practice	0	2	22	10	**34**
Total	**0**	**3**	**30**	**15**	**45**

Note:

- The cornerstone project team size shall be four (4) students.
- Students may select any one of the above prescribed projects or a project of their own choice with the prior permission of the Course instructor.
- The evaluation of the Cornerstone Project shall be based on the project implementation, oral presentation, 10-minute video presentation, report and viva voce.
- The video presentation should consist of the working procedure of the project along with contribution of each student for a minimum of 2 minutes.
- Finally, the Source code of the cornerstone Project has to be pushed into the Students GitHub repository (if applicable).

Data Mining
(Common to CSE, IT, AIML & CSE(DS))

	L	T	P	C
Course Code: 2501AI03	**2**	**0**	**1**	**3**

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

CO1: Illustrate data warehousing architectures and mining concepts for knowledge discovery.
CO2: Apply preprocessing techniques to prepare data for mining algorithms.
CO3: Build classification model using Decision tree induction.
CO4: Analyze association rule generation using Apriori and FP growth algorithms.
CO5: Apply Clustering algorithms on given data to find similarity between classes.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	2	2	2	-	1	-	-	1	1	-	2
CO2	2	2	2	-	2	-	-	1	1	-	2
CO3	2	2	3	-	1	-	-	1	1	-	2
CO4	2	2	3	-	1	-	-	1	1	-	2
CO5	2	2	3	-	1	-	-	1	1	-	2

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1	-	1
CO2	-	2
CO3	-	2
CO4	-	2
CO5	-	2

UNIT-I

Data Warehousing and Online Analytical Processing:
Data Warehouse: Basic Concepts, Data Warehouse Modeling: Data Cube and OLAP, Data Warehouse Design and Usage, Data Warehouse Implementation, Data Generalization by Attribute Oriented Induction.
Data Mining: What Is Data Mining, Why Data Mining, Data Mining Functionalities, Technologies Are Used, Major Issues in Data Mining, Architecture Of Data Mining Systems Classification Of Data Mining Systems.

Practice:
1. Illustrate KDD process and explore the Data Mining tool WEKA/KNIME for Data Preprocessing, Classification and Clustering.
2. Explain ETL process important for moving data in the data warehouse.

UNIT-II

Know Your Data : Data Objects and Attribute Types, Basic Statistical Descriptions of Data, Data Visualization, Measuring Data Similarity and Dissimilarity.
Data Preprocessing : An Overview, Data Cleaning – Data Integration and Transformation – Data Reduction – Data Discretization.

Practice:

1. Apply Statistical Descriptions and Measures of Similarity and Dissimilarity on given data
2. Apply Preprocessing for training data set related to Weather data

UNIT-III

Classification and Prediction: Classification: Basic Concepts, General Approach to Solving a Classification Problem, Decision Tree Induction, Model Overfitting, Evaluating the Performance of a Classifier, Issues Regarding Classification and Prediction

Practice:

1. Demonstrate classification rule process on given data set using id3 algorithm.
2. Demonstrate classification rule process on Student's Academic Performance Data using j48 algorithm.

UNIT-IV

Mining Frequent Patterns, Associations and Correlations: Basic Concepts and Methods, Frequent Itemset Mining Methods, Which Patterns Are Interesting?—Pattern Evaluation Methods, Association Mining to Correlation Analysis.

Practice:

1. Demonstrate Association rule generation on credit card data using apriori algorithm.
2. Demonstrate Association rule process on contact lenses. arff using FPGrowth Algorithm

UNIT – V

Cluster Analysis: Types of Data in Cluster Analysis – A Categorization of Major Clustering Methods – Partitioning Methods – Hierarchical methods – Evaluation of Clustering, Mining Spatial, Multimedia, Text and Web Data: Multidimensional Analysis and Descriptive Mining of Complex Data Objects – Spatial Data Mining – Multimedia Data Mining – Text Mining – Mining the World Wide Web.

Practice

1. Demonstrate clustering process on given dataset using simple k-means
2. Apply Hierarchical clustering process on dataset.

Additional Practice:

1. Write a program of cluster analysis using DB SCAN algorithm Python programming language
2. Demonstrate Web/Text Mining using WEKA/KNIME Tool

Capstone project:

Implement a data mining application involving data preprocessing and various data mining tasks can be achieved by focusing on a specific domain, such as retail, sales and customers, and then applying different techniques to analyse the data. This project would involve data cleaning, transformation, and various data mining tasks like clustering and association rule mining of your own data set, potentially leading to valuable insights into various area of data mining Applications.

Text Books:

1 Data Mining Concepts and Techniques, Jiawei Han, Micheline Kamber and Jian Pei Third Edition, Elsevier, ISBN: 978-9380931913
2 Introduction to Data Mining, PangNing Tan, Michael Steinbach and Vipin Kumar Pearson Education, ISBN: 978-1-292-02615-2

Reference Books:

1 Data Mining: Introductory and Advanced Topics, Margaret H. Dunham, 1st Edition Pearson Edn, ISBN: 978-8177587852
2 Data Mining Techniques and Applications An Introduction, Hongbo Du, 1st Edition, Cengage India, ISBN: 978-8131519554
3 Data Warehousing, Data Mining and OLAP, Alex Berson,.Stephen J. Smith, McGrawHill series, ISBN: 978-0070062726

Web Links:

1 https://nptel.ac.in/courses/106105174/
2 https://www.kdnuggets.com/
3 https://www.import.io/post/datascientistsvsdataanalystswhythedistinctionmatters/

Java Programming
(Common to CSE, IT, AIML & CSE(DS))

	L	T	P	C
Course Code: 2501IT06	**2**	**0**	**2**	**4**

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

CO1: Apply Java features for problem solving.
CO2: Build applications using principles of OOPs, interfaces and Packages.
CO3: Develop programs using Exception Handling to handle run-time errors.
CO4: Develop applications using multithreading for inter thread communication.
CO5: Build JDBC applications for performing CRUD operations using MySQL.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	2	1	1	-	1	-	-	-	1	2	1
CO2	2	1	3	-	1	-	-	-	1	2	1
CO3	2	3	2	-	1	-	-	-	1	2	1
CO4	2	2	2	-	1	-	-	-	1	2	1
CO5	2	2	3	-	1	-	-	-	1	2	1

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1	1	-
CO2	1	-
CO3	1	-
CO4	2	-
CO5	3	-

UNIT – I

Introduction to Java: History of Java, Features of Java Language, JVM Architecture, Java source file structure. Building Blocks of Java, Control Statements, Operators, Type Conversion, Command Line Arguments, Arrays.

Practice:

1. **Control Statements**

a. Select all the prime numbers within the range of 1 to 100.
https://leetcode.com/problems/count-primes/description/?utm_source

b. Find the sum of all even terms in the Fibonacci sequence up to the given range N.
https://leetcode.com/problems/fibonacci-number/description/

c. Check whether a given number is Armstrong or not.
https://www.hackerrank.com/contests/iiitdm-contest1/challenges/armstrong-number-2

2. Arrays

a. Sort an array of integers in ascending order.

b. Find the maximum and minimum element in an array.

c. Remove duplicate elements from an array.

UNIT – II

Concepts of Class: Class, Object, Method, Constructor, Overloading- Method and Constructor Keywords: this, static and final, String Class and its Methods, StringTokenizer, StringBuffer and StringBuilder.

Practice:

1. Class Mechanism

a. Display the details of a person. Personal details should be given in one method and the qualification details in another method.

b. Implement constructor and constructor overloading.

c. Implement method overloading.

2. Strings

a. Check if a given string is a pangram (contains every letter of the alphabet at least once).

b. Find the most frequently occurring character in a string.

c. Find all permutations of a given string.

d. Check if a given string is a anagram (Ex: CAT and ACT).

https://leetcode.com/problems/valid-anagram/description/

UNIT – III

Inheritance: Inheritance, Types of Inheritance, super and super( ), Method Overriding, Using final with inheritance, abstract with Class and Method, Dynamic Method Dispatch. Interfaces: Introduction to interface, Multiple inheritance.

Practice:

1. Inheritance & interface

https://www.hackerrank.com/challenges/java-inheritance-1/problem

https://www.hackerrank.com/challenges/java-interface/problem

a. Implement multi level Inheritance.

b. Implement multiple Inheritance.

https://www.hackerrank.com/challenges/java-interface/problem

c. Find the areas of different shapes using abstract classes.

https://www.hackerrank.com/challenges/java-abstract-class/problem

UNIT – IV

Packages: Introduction, Importing Packages and Classes, path and classpath, Access Specifiers, java.util: Collection Framework (related interfaces and classes), java.io:

FileReader, FileWriter, FileInputStream, FileOutputStream, BufferReader and BufferWriter, Wrapper Classes.

Practice:

1. Packages & java.io

a. Import and use user defined package.
https://www.geeksforgeeks.org/java/packages-in-java/

b. Illustrate the use of protected members in a package.

c. Copy Even numbers into Even.txt file and Odd Numbers into Odd.txt file.

2. Collection Framework

a. ArrayList and LinkedList

b. Iterator and Iterable

c. Comparator and Comparable

d. HashMap and TreeMap

e. HashSet and TreeSet

f. HashTable

UNIT – V

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

Multithreading: Introduction, Thread Life Cycle, Creation, Priorities, Synchronization, ITC.

Java Database Connectivity: Introduction, JDBC Architecture, Types of Drivers, JDBC Environment Setup, JDBC Steps, CRUD Operations.

Practice:

1. Exception Handling

a. Illustrate exception handling mechanism using multiple catch clauses.

b. Make use of Built-in and user-defined Exceptions in handling a run time exception.

2. Multithreading

a. Creates threads by extending Thread class. First thread display "Good Morning "every 1 sec, the second thread displays "Hello "every 2 seconds and the third display "Welcome" every 3 seconds, (Repeat the same by implementing Runnable).

b. Solve Producer-Consumer problem using synchronization.

3. JDBC

a. CRUD operations.

Capstone Project:

PROJECT 1: Student Attendance Management System
Project Implementation, Documentation and Viva Voce

- **Implementation mainly focuses on**
  - Adding student details, marking and viewing attendance.
  - JDBC operations: insert, update, search by roll number.

- Input validation (no duplicate roll numbers, days ≤ total days).
- Attendance percentage calculation.
- Menu-driven console workflow with exception handling.

- **Documentation must contain**
  - Objective & problem definition
  - Attendance table structure
  - ER diagram (Student → Attendance)
  - Module explanations: Add, Mark, View, Search
  - JDBC connection flow
  - Screenshots of sample input/output
  - Enhancements added (login, validation, normalization)

PROJECT 2: Library Management System

- **Implementation mainly focuses on**
  - Adding, searching, updating, deleting book records.
  - JDBC CRUD operations on books table.
  - Validation of price, quantity.
  - Implementing search by ID, title, and author.

- **Documentation must contain**
  - Objective of library automation
  - Books table schema
  - Flowchart for Add → View → Delete operations
  - Module descriptions
  - Exception handling (invalid ID)
  - Screenshots
  - Enhancements (issue records, author search)

PROJECT 3: Student Course Registration System

Implementation Mainly Focuses On

- Course registration, view, update, delete operations.
- Database design with student, course, registration tables.
- Validation for duplicate registration.
- JDBC connectivity and table relationships.

Documentation Must Contain

- Objective
- Table design (student, course, registration)
- ER diagram
- Module explanation: Register, Display, Edit
- SQL constraints used
- Screenshots of registration process

PROJECT 4: Online Student Result Management System

Implementation Mainly Focuses On

- Marks entry and validation.
- Automatic total, percentage, grade calculation.
- Result storage and retrieval using JDBC.
- Web-based form input using HTML.

Documentation Must Contain

- Objective
- Marks table schema
- Grade calculation logic
- HTML form screenshots
- JDBC code explanation
- Enhancements (validation, grade generation)

PROJECT 5: Employee Leave Management System

Implementation Mainly Focuses On

- Leave application submission.
- Balance update logic.
- Validations on leave days and employee ID.
- JDBC-based leave approval or rejection.

Documentation Must Contain

- Objective
- Employee & leave tables
- Flowchart: Apply → Validate → Approve
- Module explanation
- Screenshots of leave form and output

PROJECT 6: Online Shopping Cart Simulation

Implementation Mainly Focuses On

- Displaying products and adding items to cart.
- Stock validation before adding to cart.
- Bill calculation with discounts/coupons.
- JDBC for storing cart and order history.

Documentation Must Contain

- Objective
- Product, cart, order tables
- Module flow: Add → Update → Bill
- Billing logic with formulas
- Screenshots of cart process

PROJECT 7: Bus Ticket Booking System

Implementation Mainly Focuses On

- Booking, cancelling, viewing seat availability.
- Preventing double booking of same seat.
- Route table and seat allocation logic.
- JDBC insert/update operations.

Documentation Must Contain

- Objective
- Route and seat tables
- Seat allocation flowchart
- Module descriptions
- SQL queries
- Screenshots of booking/cancel pages

Text Books:

1. The Complete Reference Java, Herbert Schildt, TMH ,8th Edition, ISBN 978-1260440232.
2. Java one step ahead, Anita seth, B.L.Juneja, Oxford, First Edition. ISBN: 9780199459643.

Reference Books:

1. Introduction to java programming, by Y Daniel Liang, Seventh Edition, Pearson, ISBN: 978-8131729588
2. Core Java: An Integrated Approach, R.Nageswara Rao, Dream tech press, ISBN: 978-9351199250

Web Links:

1. https://nptel.ac.in/courses/106/105/106105191/
2. http://java.sun.com/docs/books/tutorial/

Note:

- Students must submit the Certificate of Completion offered by any Industry.
- The capstone project team size shall be four (4) students.
- Students may select any one of the above prescribed projects or a project of their own choice with the prior permission of the Course instructor.
- The SEE – Lab shall be evaluated for 50 marks based on project implementation, oral presentation, 10-minute video presentation, report and viva voce.
- The video presentation should consist of the working procedure of the project along with contribution of each student for a minimum of 2 minutes.
- Finally, the Source code of the capstone Project has to be pushed into the Students GitHub repository.

Additional Practice:

UNIT-I		
1	Finding 3-Digit Even Numbers	Link:https://leetcode.com/problems/finding-3-digit-even-numbers/description/
2	Loops	Link: https://www.hackerrank.com/challenges/30-loops/problem
3	Unique 3-Digit Even Numbers	Link:https://leetcode.com/problems/unique-3-digit-even-numbers/description/
4	Best Time to Buy and Sell Stock	Link:https://leetcode.com/problems/best-time-to-buy-and-sell-stock/description/
5	Single Number	Link: https://leetcode.com/problems/single-number/description/
6	Majority Element	Link: https://leetcode.com/problems/majority-element/description/

7	Contains Duplicate	Link: https://leetcode.com/problems/contains-duplicate/description/
8	Missing Number	Link: https://leetcode.com/problems/missing-number/description/
9	Second Largest Number	Link:https://www.naukri.com/code360/problem-details/second-largest-number_13762781
10	Java Program to Find if a Given Year is a Leap Year	Link:https://www.geeksforgeeks.org/java/java-program-to-find-if-a-given-year-is-a-leap-year/
11	Bitwise Operators	Link:https://www.geeksforgeeks.org/problems/bitwise-operators-1597314674/0
12	Program to Convert Octal Number to Binary Number	Link:https://www.geeksforgeeks.org/dsa/program-to-convert-octal-number-to-binary-number/
13	Neon Number	Link: https://www.geeksforgeeks.org/dsa/neon-number/
14	Automorphic Number	Link: https://www.geeksforgeeks.org/dsa/automorphic-number/
15	Harshad (Or Niven) Number	Link: https://www.geeksforgeeks.org/dsa/harshad-or-niven-number/
UNIT-II		
16	Longest Common Prefix	Link: https://leetcode.com/problems/longest-common-prefix/description/
17	Find the index of the first occurrence in a string	Link:https://leetcode.com/problems/find-the-index-of-the-first-occurrence-in-a-string/description/
18	Valid-Parentheses	Link: https://leetcode.com/problems/valid-parentheses/description/
19	Length of last word	Link: https://leetcode.com/problems/length-of-last-word/description/
20	Add Binary	Link: https://leetcode.com/problems/add-binary/description/
21	Isomorphic Strings	Link: https://leetcode.com/problems/isomorphic-strings/description/
22	Jewels and Stones	Link: https://leetcode.com/problems/jewels-and-stones/description/
23	Buddy Strings	Link: https://leetcode.com/problems/buddy-strings/description/
24	Reverse Only Letters	Link: https://leetcode.com/problems/reverse-only-letters/description/
25	Find Common Characters	Link: https://leetcode.com/problems/find-common-characters/description/
26	Find Pattern	Link: https://www.geeksforgeeks.org/problems/find-pattern/0
27	Largest Substring Between Two Equal Characters	Link:https://leetcode.com/problems/largest-substring-between-two-equal-characters/description/

28	Remove All Adjacent Duplicates In String	Link:https://leetcode.com/problems/remove-all-adjacent-duplicates-in-string/description
29	Permutations of given string	Link:https://www.geeksforgeeks.org/dsa/write-a-c-program-to-print-all-permutations-of-a-given-string
30	Longest Repeating Subsequence	Link: https://www.geeksforgeeks.org/dsa/longest-repeating-subsequence/
UNIT-III		
31	Java Inheritance-1	Link: https://www.hackerrank.com/challenges/java-inheritance-1/problem
32	Java Inheritance-2	Link: https://www.hackerrank.com/challenges/java-inheritance-2/problem
33	Java Abstract Class	Link: https://www.hackerrank.com/challenges/java-abstract-class/problem
34	Java Interface	Link: https://www.hackerrank.com/challenges/java-interface/problem
35	Java Method Overriding	Link: https://www.hackerrank.com/challenges/java-method-overriding/problem
36	Java Method Overriding-2 (super keyword)	Link:https://www.hackerrank.com/challenges/java-method-overriding-2-super-keyword/problem
UNIT-IV		
37	Java program to append a string in an existing file	Link: https://www.geeksforgeeks.org/java/java-program-to-append-a-string-in-an-existing-file/
38	Java End-of-file	Link: https://www.hackerrank.com/challenges/java-end-of-file/problem
39	Java Stdin and Stdout II	Link: https://www.hackerrank.com/challenges/java-stdin-stdout/problem
40	Java Output Formatting	Link: https://www.hackerrank.com/challenges/java-output-formatting/problem
41	Find Duplicate File in System	Link: https://leetcode.com/problems/find-duplicate-file-in-system/description/
UNIT-V		
42	Java Exception Handling (Try-catch)	Link: https://www.hackerrank.com/challenges/java-exception-handling-try-catch
43	Java Exception Handling	Link: https://www.hackerrank.com/challenges/java-exception-handling
44	Java Factory Pattern	Link: https://www.hackerrank.com/challenges/java-factory
45	Print in Order	Link: https://leetcode.com/problems/print-in-order/description/
46	Print Zero Even Odd	Link: https://leetcode.com/problems/print-zero-even-odd/description/

	Code chef/greek for greeks	Hacker rank	Leet code	Total
Unit-1	0	1	2	**3**
Unit-2	0	0	1	**1**
Unit-3	0	4	0	**4**
Unit-4	1	0	0	**1**
Unit-5	0	0	0	**0**
Additional Practice	10	13	23	**46**
Total	**11**	**18**	**26**	**55**

Full Stack Application Development using MERN
Common to CSE, AIML & DS
(Syllabus for the batch admitted in the year 2026)

VI Semester	L	T	P	C
Course Code: 2601AI02	2	0	2	4

Course Outcomes:

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

CO1: Develop static web pages using HTML and CSS.
CO2: Apply JavaScript for Client side validations and XML for data transportation.
CO3: Make use of Node.JS and Angular JS for developing a web server and dynamic webpages.
CO4: Utilize React JS for developing responsive web pages.
CO5: Develop web applications using PHP and MySQL/MongoDB.

Mapping of Course Outcomes with Program Outcomes:

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

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1	2	-
CO2	2	-
CO3	2	-
CO4	2	-
CO5	2	-

Unit – I: HTML and CSS

HTML 5: Introduction to Web, Overview of Web Technologies, HTML - Introduction, HTML - Need, Case-insensitivity, Platform-independency, DOCTYPE Declaration, Types of Elements, HTML Elements - Attributes, Metadata Element, Sectioning Elements, Paragraph Element, Division and Span Elements, List Element, Link Element, Character Entities, HTML5 Global Attributes, Creating Table Elements, Table Elements : Colspan/Rowspan Attributes, border, cellspacing and cellpadding attributes, Creating Form Elements, Input Elements - Attributes, Color and Date Pickers, Select and Datalist Elements, Editing Elements, Media, Iframe, Dynamic HTML.

CSS: Cascading style sheets, Levels of Style Sheets, Style Specification Formats, Selector Forms, Box Model, Conflict Resolution, CSS3.

Practice:

HTML:

1. Write a HTML program, that makes use of basic tags like <html> ,<head>, <title>, <body>, <p>, <hr>, <br>, <h1> to <h6>, <! -- --> tags and their attributes.
2. Write a HTML program, that makes use of formatting tags like <b>, <i>, <u>, , <sub>, <super>, <tt>, <pre>, <font> tags and their attributes.
3. Write a HTML program, to explain the working of lists.
   Note: It should have an ordered list, unordered list, nested lists and ordered list in an unordered list and definition lists.
4. Write a HTML program, to explain the working of hyperlinks using <a> tag and href, target Attributes.
5. Note: Use text to link →https://www.aec.edu.in/
6. Use image to link →https://www.aec.edu.in/?p=Gallery
7. Write a HTML program, to explain the working of tables by preparing a timetable. (Note: Use <caption> tag to set the caption to the table & also use cell spacing, cell padding, table border etc.).
8. Write a HTML program, to explain the working of frames, such that page is to be divided into 3 parts on either direction. (Note: first frame → image, second frame→ paragraph, third frame → hyperlink. And make sure of using "no frame" attribute such that frames to be fixed).
9. Write a HTML program, to explain the working of forms by designing Registration form. (Note: Include text field, password field, number field, date of birth field, checkboxes, radio buttons, list boxes using <select>&<option> tags, <text area> and two buttons ie: submit and reset. Use tables to provide a better view).
10. Write a HTML program, that makes use of <article>, <aside>, <figure>, <figcaption>, <footer>, <header>, <main>, <nav>, <section>, , tags.

CSS

1. Write a program to apply different types (or levels of styles or style specification formats) - inline, internal, external styles to HTML elements.
2. Write a program to apply different types of selector forms
   i. Simple selector (element, id, class, group, universal)
   ii. Combinator selector (descendant, child, adjacent sibling, general sibling)
   iii. Pseudo-class selector
   iv. Pseudo-element selector
   v. Attribute selector
3. Write a program to apply Box model.
4. Write a CSS rule that places a background image halfway down the page, tilting it horizontally. The image should remain in place when the user scrolls up or down.
5. Write a program using the following terms related to CSS:
   i. font-size ii. font-weight iii. Font-strech
   iv. text-decoration v. text-transformation vi. Text-alignment

Unit – II: Javascript and XML

JavaScript: Why we need JavaScript, What is JavaScript, Environment Setup, Working

with Identifiers, Type of Identifiers, Primitive and Non Primitive Data Types, Variables – var, let, const, Operators and Types of Operators, Types of Statements, Non - Conditional Statements, Types of Conditional Statements, If and Switch Statements, Types of Loops, Types of Functions, Declaring and Invoking Function, Arrow Function, Function Parameters, Nested Function, Built-in Functions, Variable Scope in Functions, Working With Classes, Creating and Inheriting Classes, In-built Events and Handlers, Working with Objects, Types of Objects, Predefined - String, Number, Array, Date, Math, Random, RegExp, User-defined objects - Creating Objects, Combining and cloning Objects using Spread operator, Destructuring Objects, Browser and Document Object Model, Introduction to Asynchronous Programming, Callbacks, Promises, Async and Await, Executing Network Requests using Fetch API, Creating and consuming Modules.

Working with XML: Document Type Definition (DTD), XML schemas, XSLT, XML and CSS, Document Object Model, Parsers - DOM and SAX.

Practice:

1. Write a JavaScript program, that makes use of different objects (predefined – Array, String, Math, Date, RegExp and User-defined). Note: Use atleast 5 properties and methods from each object.
2. Write a javascript to display the denomination of the amount deposited in the bank in terms of 100's, 50's, 20's, 10's, 5's, 2's & 1's. (Eg: If deposited amount is Rs.163, the output should be 1-100's, 1-50's, 1- 10's, 1-2's & 1-1's)
3. Write a JavaScript that takes a number from one text field in the range of 0-999 and display it in other text field in words. If the number is out of range, it should show "out of range" and if it is not a number, it should show "not a number" message in the result box.
4. Write a JavaScript program, that makes use of class, object, method, constructor for student information system.
5. Design a HTML having a text box and four buttons named Factorial, Fibonacci, Prime, and Palindrome. When a button is pressed an appropriate java script function should be called to display
    i. Factorial of that number
    ii. Fibonacci series up to that number
    iii. Prime numbers up to that number
    iv. Is it palindrome or not?
6. Write JavaScript programs on Event Handling
    i. Open a Window from the current window
    ii. Change color of background at each click of button or refresh of a page
    iii. Display calendar for the month and year selected from combo box
    iv. On Mouse over event
7. Write a JavaScript to validate the following fields in a registration page created in Experiment 2
    i. Name (start with alphabet and followed by alphanumeric and the length should not be less than 6 characters)
    ii. Password (it allows alphanumeric, special symbols and should not be less than 6 characters)
    iii. E-mail (should not contain invalid email addresses)
8. Write an XML file which will display the Book information which includes the following: Title of the book, Author Name, ISBN number, Publisher name, Edition, Price.

i. Write a Document Type Definition (DTD) to validate the above XML file.
ii. Write an XML Schema Definition (XSD) to validate the above XML file.
iii. Write an Extensible Stylesheet Language Transformations (XSLT) to display data in table format.

Unit – III: Node.js and MongoDB

Node.js: **Why and What Node.js, How to use Node.js, Create a web server in Node.js, Process Model, NPM, Modules - HTTP, File system, URL, Modular programming in Node.js, Restarting Node Application, Events.**

MongoDB: Introduction Module Overview, Document Database Overview, Understanding JSON, MongoDB Structure and Architecture, MongoDB Remote Management, Installing MongoDB on the local computer (Mac or Windows), Introduction to MongoDB Cloud, Create MongoDB Atlas Cluster, GUI tools Overview, Install and Configure MongoDB Compass, Introduction to the MongoDB Shell, MongoDB Shell JavaScript Engine, MongoDB Shell JavaScript Syntax, Introduction to the MongoDB Data Types, CRUD Operations on documents, Create and Delete Databases and Collections, MongoDB Queries.

Practice:
Node.js

1. How to execute different functions successfully in the Node.js platform.
2. Write a program to show the workflow of JavaScript code executable by creating web server in Node.js
3. Write a Node.js module to show the workflow of modularization of Node application.
4. Write a program to show the workflow of restarting a Node application.
5. Create a text file src.txt and add the following data to it. Mongo, Express, Angular, Node.

MongoDB
Install MongoDB and configure ATLAS.

1. Write MongoDB queries to perform CRUD operations on document using insert(), find(), update(), remove().
2. Write MongoDB queries to create and drop databases and collections

Unit – IV: Express.js and Typescript

Express.js: Express Development Environment, Defining a route, Handling Routes, Route and Query Parameters, How Middleware works, Chaining of Middlewares, Types of Middlewares, Connecting to MongoDB with Mongoose, Validation Types and Defaults, Models, CRUD Operations, API Development, Session management - Cookies, Sessions.

Typescript: Installation, Introduction, Function, Parameter Types and Return Types, Arrow Function, Function Types, Optional and Default Parameters, Rest Parameter, Creating an Interface, Duck Typing, Function Types, Extending Interfaces, Classes, Constructor, Access Modifiers, Properties and Methods, Creating and using Namespaces, Creating and using Modules, Module Formats and Loaders, Module Vs Namespace, What is Generics, What are Type Parameters, Generic Functions, Generic Constraints.ice:

Express.js

1. Implement routing using routes/route.js file.
2. Handle POST submissions, display customized error messages, perform logging
3. Write a Mongoose schema to connect with MongoDB.
4. Write a program to wrap the Schema into a Model object
5. Write a program to perform various CRUD (Create-Read-Update-Delete) operations using Mongoose library functions.
6. Include APIs based on the requirements provided - Fetch the details which is provided in the URL, Update the details based on the name which is provided in the URL and the data in the request body, Delete the details based on the name which is provided in the URL.
7. Write a program to explain session management using cookies.
8. Write a program to explain session management using sessions.

Typescript

1. Basics of TypeScript.
2. Define an arrow function inside the event handler to filter the product array with the selected product object using the productId received by the function. Pass the selected product object to the next screen.
3. Consider that developer needs to declare a function - getMobileByVendor which accepts string as input parameter and returns the list of mobiles.
4. Arrow Function.
5. Optional, Default, Rest parameters in a function.
6. Declare an interface named - Product with two properties like productId and productName with a number and string datatype and need to implement logic to populate the Product details.
7. Declare an interface named - Product with two properties like productId and productName with the number and string datatype and need to implement logic to populate the Product details.
8. Declare an interface with function type and access its value.
9. Declare a productList interface which extends properties from two other declared interfaces like Category, Product as well as implementation to create a variable of this interface type.
10. Consider the Mobile Cart application, Create objects of the Product class and place them into the productlist array.
11. Declare a class named - Product with the below-mentioned declarations: (i) productid as number property (ii) Constructor to initialize this value (iii) getProductId method to return the message "Product id is <<id value>>".

12. Create a Product class with 4 properties namely productId, productName, productPrice, productCategory with private, public, static, and protected access modifiers and accessing them through Gadget class and its methods.
13. Create a Product class with 4 properties namely productId and methods to setProductId() and getProductId().
14. Create a namespace called ProductUtility and place the Product class definition in it.
15. Import the Product class inside productlist file and use it.
16. Consider the Mobile Cart application which is designed as part of the functions in a module to calculate the total price of the product using the quantity and price values and assign it to a totalPrice variable.
17. Create a generic array and function to sort numbers as well as string values.

Unit – V: React JS

React JS – Introduction, Displaying "Welcome React", Introducing JSX, Rendering Elements, Components and Props, State and Lifecycle, Handling Events, Conditional Rendering, Lists and Keys, Forms, Lifting State Up, Composition vs Inheritance, Thinking in React, Hooks, API Integration

Practice:

1. Create a simple React application that displays the message "Welcome React" on the webpage.
2. Develop a React component that uses JSX syntax to display static and dynamic content.
3. Create and render elements dynamically.
4. Design a component that receives and displays student information through props and states.
5. Implement simple component and stateful component.
6. Implement form validation using events.
7. Implement conditional rendering.
8. Implement rendering lists and keys.
9. Implement routing to different pages using react router.
10. Implement hooks.
11. Implement sharing data between components
12. Implement to-do list application.
13. Implement API integration – CRUD operations using MySQL

Capstone Projects:

Project 1: Student Management System

Implementation mainly focuses on CRUD operations using MongoDB, Express, React, Node.js

Project 2: Online Food Ordering System

Implementation mainly focuses on menu display, cart, orders, user login, admin panel.

Project 3: Personal Portfolio with Admin Dashboard

Implementation mainly focuses on Edit portfolio sections via a protected admin panel.

Project 4: To-Do List Application
Implementation mainly focuses on add, edit, update and delete tasks.

Project 5: Event Registration Portal
Implementation mainly focuses on users register for events; admin can view/manage registrations.

Project 6: E-Commerce Product Catalog
Implementation mainly focuses on Product listings, categories, product details, cart basics.

Project 7: College Information & Admission Portal
Implementation mainly focuses on Forms, tables, student admission data, admin management.

Project 8: Blog Posting Platform
Implementation mainly focuses on users create posts, add images, edit, comment, delete.

Project 9: Movie Review and Rating System
Implementation mainly focuses on users can review movies, give ratings, view reviews list.

Project 10: Employee Record Management
Implementation mainly focuses on Employee details, salary, department, search filters.

Project 11: Notes Application
Implementation mainly focuses on Create/edit notes, categorize them, private notes per user.

Project 12: Online Quiz Application
Implementation mainly focuses on admin adding questions, users take quiz, store scores.

Project 13: Library Book Management System
Implementation mainly focuses on add books, issue/return system, student records, dashboards.

Project 14: Hospital Appointment Booking System
Implementation mainly focuses on users book appointments, doctors manage schedules.

Note: Documentation must contain Problem Definition, Objectives, Analysis & Design, Implementation and Sample Code, Output screens, Conclusion & Enhancement.

Text Books:

1. Programming the World Wide Web, 7th Edition, Robet W Sebesta, Pearson
2. Pro Mean Stack Development, 1st Edition, ELadElrom, Apress O'Reilly.
3. React Explained, 2020 Edition, Zac Gordon, OSTraining.
4. MongoDB – The Definitive Guide, 2nd Edition, Kristina Chodorow,O'Reilly.

Reference Books:

1. Web Technologies, HTML, JavaScript, PHP, Java, JSP, XML and AJAX, Black book,1st Edition, Dream Tech.
2. An Introduction to Web Design, Programming, 1st Edition, Paul S Wang, Sanda S Katila, Cengage Learning.

Web Links:

1. https://www.reactjs.org/docs/getting-started.html
2. https://www.university.mongodb.com/

Note:

- The cornerstone project team size shall be four (4) students.
- Students may select any one of the above prescribed projects or a project of their own choice with the prior permission of the Course instructor.
- The evaluation of the Cornerstone Project shall be based on the project implementation, oral presentation, 10-minute video presentation, report and viva voce.
- The video presentation should consist of the working procedure of the project along with contribution of each student for a minimum of 2 minutes.
- Finally, the Source code of the cornerstone Project has to be pushed into the Students GitHub repository (if applicable).

Language Processors
(Common to AIML & CSE(DS))

	L	T	P	C
Course Code: 2501CS09	**2**	**1**	**0**	**3**

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

CO1: Explain the equivalence between NFA, DFA and Regular Expressions
CO2: Illustrate the concepts of Context Free Grammars, Push Down Automata and Turing Machines
CO3: Analyze the phases in compilation process.
CO4: Apply different parsing algorithms to develop parsers for a given grammar
CO5: Discuss code generation and code optimization techniques

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	3	-	1	-	-	-	-	-	-	-	-
CO2	1	2	1	-	-	-	-	-	-	-	-
CO3	1	2	1	-	-	-	-	-	-	-	-
CO4	3	-	-	1	-	-	-	-	-	-	-
CO5	2	2	-	2	-	-	-	-	-	-	-

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1	2	-
CO2	2	-
CO3	2	-
CO4	2	-
CO5	2	-

UNIT-I

Finite Automata and Regular Expressions: Formal Languages and Regular expressions, Deterministic and Non Deterministic Finite Automata, Finite Automata with ε moves, Equivalence of NFA and DFA, Minimization of Finite Automata, Applications of Finite Automata.

UNIT-II

Grammars , PDA and Turing Machines: Chomsky hierarchy, Properties of regular sets, Pumping Lemma for regular languages, ContextFree Grammars – Derivation trees, Ambiguous and unambiguous grammars , Left recursion, Left factoring, Pushdown Automata and ContextFree Languages, Turing machines (TM) – Turing Machine constructions – Variations of TMs.

UNIT-III

Phases of Compiler and Lexical Analyzer : Compilers Analysis of the source program The phases of a compiler Cousins of the compiler Compiler construction tools Lexical Analysis The role of the lexical analyzer Input buffering Specification of tokens Recognition of tokens A language for specifying lexical analyzers Design of a lexical analyzer.

UNIT-IV

Syntax Analysis and Syntax Directed Translation: The role of the parser Context free grammars Top down parsing Bottom up parsing Operator precedence parsing – automatic construction of efficient parser – predictive parser LR parsers Parser generators. Syntax directed definitions Construction of syntax trees Bottom up evaluation of Sattributed definitions Lattributed definitions Analysis of syntax directed definitions.

UNIT – V

Intermediate Code Generation and Code Generation: Intermediate languages Declarations Assignment statements Boolean expressions Back patching Procedure calls. Issues in the design of a code generator Runtime storage management Basic blocks and flow graphs The DAG representation of Basic Block Next use information simple code generator Register allocation and assignment –Code Optimization Peephole optimization Generating code from DAGs.

Text Books:

1 Introduction to Automata Theory, Languages and Computation, J. E. Hopcroft, R. Motwani and J. D. Ullman, 3rd Edition, Pearson, 2008, ISBN: 978-8131720479
2 Compilers Principles, Techniques And Tools, Alfred V Aho, J. D. Ullman, Ravi Sethi, PEA, Second Edition, 2007, ISBN: 978-9357054119

Reference Books:

1 Theory of Computer Science-Automata, Languages and Computation, K. L. P. Mishra and N. Chandrasekharan, PHI, 3rd Edition, ISBN: 978-8120329683
2 Elements of Theory of Computation, Lewis H.P. & Papadimitriou C.H., Pearson /PHI, ISBN: 978-0132624787
3 Theory of Automata, Languages and Computation, Rajendra Kumar, McGraw Hill, ISBN: 978-0070702042
4 Compiler Construction-Principles and Practice, Kenneth C Louden, Cengage Learning, ISBN: 978-0534939724
5 The Compiler Design Handbook, Y.N. Srikant and Priti Shankar, CRC Press, 2nd ed, ISBN: 978-1420043822

Web Links:

1 http://nptel.ac.in/courses/111103016
2 https://nptel.ac.in/courses/106/104/106104148/
3 https://nptel.ac.in/courses/106/108/106108113/
4 https://nptel.ac.in/courses/106/104/106104123/
5 https://nptel.ac.in/courses/106/104/106104072/

Agile Software Engineering
(Common to CSE, IT, AIML & CSE(DS))

	L	T	P	C
Course Code: 2501IT07	**2**	**0**	**1**	**3**

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

- **CO1:** Apply software engineering practices and evaluate different software development models to design and manage the life cycle of a Student Management System (SMS).
- **CO2:** Analyze the principles of Agile methodologies and compare them with traditional models by transitioning a retail software company from Waterfall to Agile.
- **CO3:** Evaluate the impact of Agile processes on requirements engineering and manage unstable requirements for a social media platform project.
- **CO4:** Design and implement a comprehensive testing strategy using TDD and test automation tools in an Agile project.
- **CO5:** Apply Agile design principles and execute refactoring techniques to develop maintainable and scalable software systems.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	2	1	-	-	2	-	-	-	-	-	-
CO2	3	2	1	1	-	-	-	2	2	2	1
CO3	2	1	1	-	-	-	-	-	-	2	1
CO4	3	2	2	1	-	-	-	2	2	2	1
CO5	1	2	-	-	-	-	-	-	-	-	-

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1	1	-
CO2	2	-
CO3	1	1
CO4	2	2
CO5	1	1

UNIT-I

Introduction to Software Engineering: Define Software Engineering, The Software Process, Software Engineering Practice, Software Myths

Software Life Cycle models and Processes: Software Process, Process Classification, Phased Development Life Cycle, Software Development Process Models – The Waterfall, Spiral and Evolutionary Models, Software Project Management Process, Project Planning and Estimation, Gathering Requirements, Software Design, Software Testing, Software Quality and Reliability.

Case Study: Developing a Student Management System (SMS)

UNIT-II

Fundamentals of Agile Methodology: 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, Agile Drivers, Capabilities and Values

Case Study: Implementing Agile Methodology in a Software Development Project

UNIT-III

Agile Processes: Lean production - SCRUM, Crystal, Feature Driven Development, Adaptive Software Development, and Extreme Programming: Method overview – lifecycle – work products, roles and practices.

Agility And Requirements Engineering: Impact of Agile Processes in RE, Current Agile Practices, Overview of RE Using Agile, Managing Unstable Requirements, Requirements Elicitation

Case Study: Agile Processes and Requirements Engineering in a Healthcare Software Project

UNIT-IV

Agile Testing: The Agile lifecycle and its impact on testing, Test-Driven Development (TDD), xUnit framework and tools for TDD, Testing user stories - acceptance tests and scenarios, Planning and managing testing cycle, Exploratory testing, Risk based testing, Regression tests, Test Automation, Tools to support the Agile tester.

Case Study: Implementing Test-Driven Development (TDD) for a Healthcare Application

UNIT – V

Agile Software Design and Development: Agile design practices, Role of design Principles including Single Responsibility Principle, Open Closed Principle, Liskov Substitution Principle, Interface Segregation Principles, Dependency Inversion Principle in Agile Design, Need and significance of Refactoring, Refactoring Techniques, Continuous Integration, Automated build tools, Version control.

Case Study: Refactoring an E-Commerce Backend Using Agile Design Principles

Capstone Project:

Students shall choose a real-time software application (e.g., e-commerce backend, SMS, food delivery app) and apply Agile design principles, perform refactoring, implement TDD, and use version control and CI tools. This project should be implemented as a Capstone project and submitted at the end of the course.

Text Books:

1 Software Engineering – Concepts and Practices: Ugrasen Suman, Cengage Learning, ISBN: 9789390555475.

2 Agile Software Development, Current Research and Future Directions‖, Springer-Verlag Berlin Heidelberg, Dingsoyr, Torgeir, Dyba, Tore, Moe, Nils Brede (Eds.), ISBN-10. 3642125743.

Reference Books:

1 Software Engineering, Lan Sommerville, Pearson, Ninth Edition, ISBN: 978-9332582699.
2 Agile Management for Software Engineering: Applying the Theory of Constraints for Business Results‖, Prentice Hall, David J. Anderson; Eli Schragenheim, ISBN: 978-0131424609

Web Links:

1 https://www.tutorialspoint.com/software_engineering/
2 https://www.sciencedirect.com/science/article/pii/S0950584916303329

Operating Systems
(Common to CSE, IT, AIML & CSE(DS))

	L	T	P	C
Course Code: 2501CS13	**2**	**0**	**1**	**3**

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

CO1: Describe basic concepts of operating systems and its structure.
CO2: Analyse various issues related to inter process communication like process scheduling, resource management and deadlocks.
CO3: Interpret the issues and challenges of memory management.
CO4: Illustrate concepts of Disk management and file system implementation.
CO5: Explain issues related to protection and security mechanisms.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	3	-	-	-	-	-	-	1	1	-	-
CO2	1	3	-	1	-	-	-	2	2	-	1
CO3	1	3	-	1	-	-	-	1	2	-	1
CO4	1	3	-	1	-	-	-	1	1	-	1
CO5	2	2	-	-	-	-	-	1	1	-	-

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1	1	-
CO2	2	-
CO3	2	-
CO4	2	-
CO5	2	-

UNIT-I

Operating Systems Overview: Operating system functions, Operating system structure, Operating systems operations, Computing environments, OpenSource Operating Systems.
System Structures: Operating System Services, User and OperatingSystem Interface, systems calls, Types of System Calls, system programs, operating system structure, System Boot.

Case Study: The above topics are discussed as case study in Windows/ Unix OS.
Practice:

1. Basic Linux commands
2. Implementation of System calls
3. Multi-programming Memory management Implementation of fork (), wait (), exec() and exit (), System calls

UNIT-II

Process Concept: Process scheduling, Operations on processes, Interprocess communication.
Process Scheduling: Basic concepts, Scheduling criteria, Scheduling algorithms, Multiple processor scheduling, Thread scheduling.
Multithreaded Programming: Multithreading models, Thread libraries, Threading issues

Case Study: The above topics are discussed as case study in Windows/ Unix OS.
Practice:

1. Simulate the following CPU scheduling algorithms:(a) FCFS (b) SJF
2. Simulate the following CPU scheduling algorithms:(a) Priority (b) Round Robin

UNIT-III

Interprocess Communication: Race conditions, Critical Regions, Mutual exclusion with busy waiting, Sleep and wakeup, Semaphores, Mutexes, Monitors, Message passing, Classical IPC Problems Dining philosophers problem, Readers and writers problem.
Deadlocks: Resources, Conditions for resource deadlocks, Ostrich algorithm, Deadlock detection and recovery, Deadlock avoidance, Deadlock prevention.

Case Study: The above topics are discussed as case study in Windows/ Unix OS.
Practice:

1. Simulate Bankers Algorithm for Dead Lock Avoidance
2. Write a C program that illustrates two processes communicating using shared memory
3. Write C program to create a thread using pthreads library and let it run its function.
4. Write a C program to illustrate concurrent execution of threads using pthreads library

UNIT-IV

Memory Management Strategies: Introduction, Swapping, Contiguous memory allocation, Paging, Segmentation.
Virtual Memory Management: Introduction, Demand paging, Copy onwrite, Page replacement, Frame allocation, Thrashing, Memorymapped files, Kernel memory allocation.
File Systems: Files, Directories, File system implementation, management and optimization.

Case Study: The above topics are discussed as case study in Windows/ Unix OS.
Practice:

1. Simulate the Multiprogramming with a fixed number of tasks (MFT)
2. Simulate the Multiprogramming with a variable number of tasks (MVT)
3. Simulate the FIFO page replacement algorithm
4. Simulate the LRU page replacement algorithm
5. Simulate the following File allocation strategies (a) Sequenced (b) Indexed (c) Linked

UNIT – V

Secondary Storage Structure: Overview of disk structure, and attachment, Disk scheduling, RAID structure, Stable storage implementation.
System Protection: Goals of protection, Principles and domain of protection, Access matrix, Access control, Revocation of access rights.
System Security: Introduction, Program threats, System and network threats

Practice:

1. The above topics are discussed as case study in Windows/ Unix OS.

Text Books:

1	Operating System Concepts, Abraham Silberschatz, Peter B Galvin and Greg Gagne, John Wiley and Sons Inc, 12th Edition, ISBN: 978-0470128725
2	Operating Systems, William Stallings, Internals and Design Principles. Prentice-Hall, 7th Ed, ISBN: 978-9332518803

Reference Books:

1	Modern Operating Systems, Tanenbaum A S, Pearson Education, 3rd Edition, ISBN: 978-0136006633
2	Operating Systems A Concept Based Approach, Dhamdhere D M, Tata McGraw-Hill, 3rd edition, ISBN: 978-1259005589
3	Operating Systems, Nutt G, Pearson Education, 3rd edition, ISBN: 978-8131723593

Web Links:

1	https://archive.nptel.ac.in/courses/106/106/106106144/
2	https://archive.nptel.ac.in/courses/106/105/106105214/
3	https://www.coursera.org/learn/iot/lecture/MrgxS/lecture-3-1-operating-systems
4	http://www.geeksforgeeks.org/operating-systems/
5	https://in.udacity.com/auth?next=/course/introduction-to-operating-systems--ud923

Computer Networks

(Common to CSE, IT, AIML & CSE(DS))

	L	T	P	C
Course Code: 2501CS07	**2**	**0**	**1**	**3**

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

CO1: Describe network topologies, reference models and media for data transmission
CO2: Analyze error and flow control issues in data link layer
CO3: Classify MAC protocols and channelization techniques
CO4: Apply routing algorithms and congestion control techniques for effective data transmission
CO5: Analyze protocols Transport and Application Layers

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	3	1	-	-	-	-	-	-	-	-	-
CO2	-	2	-	-	-	-	-	-	-	-	-
CO3	-	2	-	-	-	-	-	-	1	1	-
CO4	1	2	2	-	-	-	-	-	2	-	1
CO5	-	1	-	-	1	-	-	-	-	-	1

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1	1	-
CO2	2	-
CO3	2	-
CO4	2	-
CO5	2	-

UNIT-I

Network Overview: Interfaces, protocols and services, connectionoriented and connectionless services, OSI & TCP/IP Reference Models. Local Area Networks: Topologies star, bus, ring, media access control deterministic and probabilistic, IEEE 802.x. wireless networking.

Practice:

1. Study of Network devices in detail and connect the computers in Local Area Network.

UNIT-II

Data Link Protocols: framing and data transparency, error detection & correction, flow control.

Practice:

1. Write a Program to implement the data link layer farming methods such as i) Character stuffing ii) bit stuffing.

UNIT-III

Media Access Control: Random Access: ALOHA, Carrier sense multiple access (CSMA), CSMA with Collision Detection, CSMA with Collision Avoidance
Controlled Access: Reservation, Polling, Token Passing
Channelization: Frequency division multiple Access (FDMA), Time division multiple access (TDMA), Code division multiple access (CDMA).

Practice:

1. Write a Program to implement Sliding window protocols

UNIT-IV

Routing Algorithms: The Optimality principle, Shortest path, Flooding, Distance vector, Link state, Hierarchical.
Congestion Control algorithms: General principles of congestion control, Congestion prevention polices, Approaches to Congestion Control, Traffic Aware Routing, Admission Control, Traffic Throttling, Load Shedding.
Traffic Control Algorithms: Leaky bucket & Token bucket.

Practice:

1. Programs to implements routing protocols like shortest path.

UNIT – V

IP Protocols: IP Addressing, IP & ICMP.
Transmission Control Protocol: UDP & TCP.
Application Layer: World Wide Web, HTTP, Electronic mail, Architecture, web based mail, email security, TELENET, local versus remote Logging.
Domain Name System: Name Space, DNS, SNMP.

Practice:

1. NS2 Simulator i. NS2 Simulator Introduction
   ii. Simulate to Find the Number of Packets Dropped
   iii.Simulate to Find the Number of Packets Dropped by TCP/UDP

Capstone Project:

Simulate and implement data link layer protocols and network routing strategies using NS2 (Network Simulator 2). It includes configuring network topologies, setting up routing protocols like AODV, DSDV, and OLSR, and analyzing their performance through metrics such as throughput, delay, and packet delivery ratio. Additionally, explore the simulation of data link layer functionalities, including MAC addressing and error detection. You may use NS2's capabilities, that insights into the behaviour of network protocols, aiding in the design and optimization of efficient communication systems.
Text Books:

1 Computer Networks — Andrew S Tanenbaum and David J Wetherall, Pearson Education, 5th Edition. ISBN: 978-0132126953
2 Data Communications and Networking – Behrouz A.Forouzan, McGraw Hill Education, 5th Edition ISBN: 978-1259064753

Reference Books:

1 Data Communications and Networks- Achut S Godbole, AtulKahate, McGraw-Hill Education (India) Pvt Limited, 2nd Edition. ISBN: 978-0-07-123110-7
2 Computer Networks, Mayank Dave, CENGAGE India, 6th Edition ISBN: 978-8131509869

Web Links:

1 https://nptel.ac.in/courses/106105081
2 https://www.coursera.org/learn/fundamentals-network-communications

Machine Learning
(Common to CSE, IT, AIML & CSE(DS))

	L	T	P	C
Course Code: 2501AI05	**2**	**0**	**2**	**4**

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

CO1: Outline the Concepts of Machine Learning and Statistical Learning
CO2: Build Regression and Classification models for given data.
CO3: Apply Instance based Learning techniques and SVM techniques linear and nonlinear data.
CO4: Apply clustering techniques on high dimensional data to group the similar entities.
CO5: Make use of ensemble learning techniques to improve the performance of a model.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	2	2	-	-	2	2	-	1	1	-	-
CO2	2	2	3	-	2	2	-	1	1	-	-
CO3	3	2	3	-	2	-	-	1	1	-	2
CO4	3	2	3	-	2	-	-	1	1	-	-
CO5	2	2	2	-	2	-	-	1	1	-	2

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1	-	1
CO2	-	2
CO3	-	2
CO4	-	2
CO5	-	2

UNIT-I

INTRODUCTION: Machine Learning, Types of Machine Learning Systems, Main Challenges of Machine Learning. Statistical Learning: Introduction, Supervised and Unsupervised Learning, The Learning Problem, Feasibility of Learning.

Practice:
1. Installation of Jupitar/Spider notebook and working on basic commands.
2. Loading and apply key preprocessing techniques on the dataset and also analyse the dataset.

UNIT-II

Supervised Learning(Regression/Classification):Basic Methods:
Concept Learning: General-to-Specific Hypotheses Ordering, Find-S and Candidate Elimination Algorithm, Version Space, and Inductive Bias.

Bayesian Learning: Probability Overview, MLE and MAP Estimates, Gaussian Naive Bayes Classifier, Bayesian Networks.
Instance-based Learning: k-Nearest Neighbour (kNN) Classifier, Voronoi Diagram and Distance-Weighted kNN, Distance Metrics and Curse of Dimensionality.

Practice:

1. Implement and demonstrate the FIND-S algorithm for finding the most specific hypothesis based on a given set of training data samples. Read the training data from a .CSV file.
2. For a given set of training data examples stored in a .CSV file, implement and demonstrate the Candidate Elimination algorithm to output a description of the set of all hypotheses consistent with the training examples.
3. Exercises to solve the real-world problems using Binary Classifier
4. Write a program to implement k-Nearest Neighbour algorithm to classify the iris data set. Print both correct andwrong predictions.

UNIT-III
Linear Models and Regression: Linear Classification, Linear Regression, Non-linear Transformation, Logistic Regression.
Support Vector Machines: Decision Boundary and Support Vector: Optimization and Primal-Dual Problem, Soft Margin and Non-linear Decision Boundary, Kernel Functions and Radial Basis Functions introduction.

Practice:

1. Exercises to solve the real-world problems using Linear Regression
2. Exercises to solve the real-world problems using Logistic Regression
3. Consider Patient Dataset. Apply linear classification technique(SVM) to identify the rate of heart patients.

UNIT-IV
Classifier / Hypothesis Evaluation: Accuracy, Precision, Recall and F-Measures, Scores, Sampling, Bootstrapping and ROC, Hypotheses Testing and Cross-validation.
Unsupervised Learning:
Clustering: Partitional Clustering and Hierarchical Clustering, Cluster Types, Attributes and Salient Features, Hierarchical and Density-based Clustering Algorithms, Inter and Intra Clustering Similarity, Cohesion and Separation. MST and DBSCAN Clustering Algorithms.

Practice:

1. Develop a program for Bias, Variance, Remove duplicates , Cross Validation
2. Write a program to implement One-hot Encoding.
3. Write a program to implement Categorical Encoding.

UNIT – V
Ensemble Learning: Bagging and Boosting, Adaboost and Random Forest
Computational Learning Theory: Error and Noise Formalisms, Training vs. Testing, Theory of Generalization, PAC Learnability and VC Dimensions, Overfitting, Regularization and Validation.

Practice:

1. Write a program to demonstrate the working of Random Forest classifier. Use appropriate dataset for Random Forest Classifier.
2. Use a sentiment analysis dataset from Twitter or other social media platforms, available on platforms like Kaggle and apply bagging and boosting techniques for prediction.

Additional Practice:

1. Assuming a set of documents that need to be classified, use the naïve Bayesian Classifier model toper form this task. Built-in Java classes/API can be used to write the program. Calculate the accuracy, precision, and recall for your data set.
2. Apply EM algorithm to cluster a Heart Disease Data Set. Use the same data set for clustering using k Means algorithm. Compare the results of these two algorithms and comment on the quality of clustering. You can add Java/Python ML library classes/API in the program.

Corner Stone Project:

PROJECT 1: Intelligent Loan Risk Assessment System (End-to-End ML Pipeline)

Project Implementation, Documentation and Viva Voce

• Implementation mainly focuses on

- Collecting and integrating multiple datasets (loan + financial history)
- Advanced preprocessing (missing values, outliers, feature engineering)
- Implementing multiple models (Logistic Regression, Random Forest, XGBoost)
- Handling imbalanced data (SMOTE / resampling techniques)
- Model evaluation using ROC-AUC, F1-score, cross-validation
- Building a simple UI (Streamlit/Web app) for predictions
- Model comparison and optimization

• Documentation must contain

- Objective & real-world problem definition
- Dataset merging and preprocessing pipeline
- Feature engineering techniques
- Model comparison and evaluation metrics
- System architecture diagram
- Screenshots (UI + outputs)
- Enhancements (hyperparameter tuning, deployment)

PROJECT 2: Smart Disease Prediction System with Model Ensemble

• Implementation mainly focuses on

- Using medical datasets (heart disease/diabetes/multi-disease)
- Data preprocessing and feature selection

- Applying multiple classifiers (SVM, kNN, Random Forest, Gradient Boosting)
- Ensemble learning (Voting / Stacking models)
- Model evaluation using confusion matrix, ROC curves
- Explainability (feature importance / SHAP basics)
- Web-based prediction interface

• Documentation must contain

- Objective
- Dataset and preprocessing steps
- Model architecture and ensemble approach
- Evaluation metrics and comparison
- Screenshots (prediction system)
- Enhancements (multi-disease prediction)

PROJECT 3: Real-Time Sentiment Analysis System with Web Scraping

• Implementation mainly focuses on

- Scraping live data (Twitter/news using APIs or scraping tools)
- Text preprocessing (stopwords, stemming, vectorization – TF-IDF)
- Training models (Naïve Bayes, Logistic Regression, LSTM optional)
- Real-time sentiment prediction
- Visualization dashboard (charts for sentiment trends)
- Handling large-scale streaming data

• Documentation must contain

- Objective
- Data collection methodology (APIs/web scraping)
- NLP preprocessing pipeline
- Model building and evaluation
- Dashboard explanation
- Screenshots
- Enhancements (real-time alerts, topic modeling)

PROJECT 4: Heart Disease Prediction using SVM

• Implementation mainly focuses on

- Loading patient dataset
- Applying Support Vector Machine (SVM)
- Data preprocessing and feature scaling
- Classification of patients (disease/no disease)
- Model evaluation using confusion matrix

• Documentation must contain

- Objective
- Dataset description
- SVM working principle
- Evaluation metrics
- Output results
- Screenshots

- Enhancements (kernel comparison)

PROJECT 5: Customer Segmentation using Clustering

• Implementation mainly focuses on

- Loading customer dataset
- Applying K-Means clustering
- Identifying groups based on features
- Visualizing clusters using plots
- Comparing clustering results

• Documentation must contain

- Objective
- Dataset details
- Clustering technique explanation
- Cluster visualization
- Results interpretation
- Screenshots
- Enhancements (DBSCAN comparison)

PROJECT 6: Fake News Detection System using Machine Learning

• Implementation mainly focuses on

- Dataset collection (news datasets / scraping)
- Text preprocessing (tokenization, TF-IDF, n-grams)
- Applying classification models (Logistic Regression, SVM, Random Forest)
- Model evaluation using precision, recall, F1-score
- Comparing multiple models
- Building a simple interface for prediction

• Documentation must contain

- Objective
- Dataset description
- NLP pipeline
- Model comparison
- Evaluation metrics
- Screenshots
- Enhancements (deep learning models)

PROJECT 7: Sentiment Analysis using Ensemble Learning

• Implementation mainly focuses on

- Collecting dataset (Twitter/Kaggle)
- Text preprocessing
- Applying Bagging and Boosting techniques
- Training ensemble models (Random Forest / AdaBoost)
- Predicting sentiment (positive/negative)

• Documentation must contain

- Objective
- Dataset source
- Preprocessing steps
- Ensemble methods explanation
- Performance comparison
- Screenshots
- Enhancements (real-time data analysis)

PROJECT 8: Model Evaluation and Comparison System

• Implementation mainly focuses on

- Implementing multiple models (kNN, SVM, Regression)
- Performing cross-validation
- Calculating accuracy, precision, recall, F1-score
- Comparing model performance
- Visualizing results using graphs

• Documentation must contain

- Objective
- Models used
- Evaluation metrics explanation
- Comparison results
- Graphs and charts
- Screenshots
- Enhancements (hyperparameter tuning)

Text Books:

1 Hands On Machine Learning with ScikitLearn, Keras, and TensorFlow, 2nd Edition, O'Reilly Publications, 2019, ISBN: 978-1492032649
2 Data Science and Machine Learning Mathematical and Statistical Methods,Dirk P. Kroese, Zdravko I. Botev, Thomas Taimre, Radislav Vaisman, 2019, ISBN: 978-1138492530

Reference Books:

1 Machine Learning Probabilistic Approach, Kevin P. Murphy, MIT Press, ISBN: 9780262018029
2 Machine Learning Tools, Dr Muddafa Maruli Krishna, Dr A Vanathi, Final Thawaksr, Dr Lokesh P. Gangani, publisher: Book Rivers, ISBN: 978-93-5515-730-0

Web Links:

1 https://www.deeplearning.ai/machinelearningyearning/
2 https://www.cse.huji.ac.il/~shais/UnderstandingMachineLearning/index.html
3 https://onlinecourses.nptel.ac.in/noc21_cs24/preview

Note:

- Students must submit the Certificate of Completion offered by any Industry.
- The capstone project team size shall be four (4) students.
- Students may select any one of the above prescribed projects or a project of their own choice with the prior permission of the Course instructor.
- The SEE – Lab shall be evaluated for 50 marks based on project implementation, oral presentation, 10-minute video presentation, and viva voce.
- The video presentation should consist of the working procedure of the project along with contribution of each student for a minimum of 2 minutes.

Advanced Data Structures & Algorithm Analysis
(Common to AIML, CSE, CSE-DS & IT)

	L	T	P	C
Course Code: 2501CS10	**2**	**0**	**1**	**3**

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

CO1: Analyze the algorithms efficiency and hashing techniques for searching.
CO2: Illustrate the concepts of search trees and operations of Priority Queues.
CO3: Analyze the algorithms of Divide & Conquer and Greedy paradigms.
CO4: Analyze the algorithms of Dynamic Programming paradigm.
CO5: Design the algorithms of Backtracking and Brach & Bound paradigms.

Mapping of Course Outcomes with Program Outcomes:

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

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1	2	-
CO2	2	-
CO3	2	-
CO4	2	-
CO5	2	-

UNIT – I

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

Hashing: Introduction to Static Hashing, Hash Tables, Hash Functions, Different Hash Functions, Collision Resolution Techniques, Dynamic Hashing.

Practice:

1. Write a program to implement collision resolution techniques of Hash data structure.

UNIT – II

AVL Trees – Creation, Insertion, Deletion operations and Applications.
B-Trees – Creation, Insertion, Deletion operations and Applications
Priority Queues (Heaps): Introduction, Binary Heaps-Model and Simple Implementation, Basic Heap Operations, Other Heap Operations, Applications of Priority

Queues

Practice:

1. Write a program to implement AVL tree operations
2. Write a program to implement Max heap and min heap operations

UNIT – III

Divide and Conquer: The General Method, Binary Search, Merge Sort, Quick Sort.

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

Practice:

1. Write a program to sort the given list of elements using Merge sort technique using divide and conquer approach.
2. Write a program to implement Single Source Shortest Paths using greedy approach.

UNIT – IV

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

Practice:

1. Write a program to implement 0/1 Knapsack problem using Dynamic Programming approach.
2. Write a program to implement all pairs shortest path problem using Dynamic Programming approach.

UNIT – V

Backtracking: The General Method, N-Queens Problem, Sum of Subsets problem, Graph Coloring, Hamiltonian cycles.

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

Practice:

1. Write a program to implement N-Queens problem using backtracking algorithm.
2. Write a program to implement Travelling Salesperson problem using branch and bound approach.

Text Books:

1. Fundamentals of Data Structures in C++, Horowitz, Ellis; Sahni, Sartaj; Mehta, Dinesh, 2nd Edition Universities Press
2. Computer Algorithms in C++, Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran,2nd Edition University Press.

Reference Books:

1. Data Structures and program design in C, Robert Kruse, 2nd Edition Pearson Education Asia.
2. An introduction to Data Structures with applications, Trembley& Sorenson, McGraw Hill
3. The Art of Computer Programming, Vol.1: Fundamental Algorithms, Donald E Knuth, Addison - Wesley.
4. Data Structures using C & C++: Langsam, Augenstein &Tanenbaum, Pearson

Web Links:

1. https://archive.nptel.ac.in/courses/106/102/106102064/
2. https://archive.nptel.ac.in/courses/106/106/106106131/
3. http://peterindia.net/Algorithms.html

Deep Learning
(Common to CSE, IT, AIML & CSE(DS))

	L	T	P	C
Course Code: 2501AI11	**2**	**0**	**2**	**4**

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

CO1: Demonstrate the basic statistical concepts in deep learning.
CO2: Explain architecture and mathematical foundation for various deep neural networks.
CO3: Illustrate the challenges and optimization strategies in Deep Learning.
CO4: Build a convolutional neural network using different activation functions.
CO5: Build and train RNN and LSTMs using sequence modelling.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	-	2	1	-	2	-	-	1	1	-	3
CO2	-	2	1	-	2	-	-	1	1	-	3
CO3	-	2	-	3	1	-	-	1	1	-	1
CO4	2	2	3	-	1	-	-	1	1	-	-
CO5	1	2	3	-	1	-	-	1	1	-	-

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1		2
CO2		2
CO3		2
CO4		2
CO5		2

UNIT-I
DEEP LEARNING CONCEPTS: Fundamentals about Deep Learning, Perception Learning Algorithms, Probabilistic modelling, Early Neural Networks, How Deep Learning different from Machine Learning, Scalars. Vectors, Matrixes, Higher Dimensional Tensors, Manipulating Tensors. Vector Data, Hyper parameters Vs Parameters - validation sets - Estimators, Bias, Variance , Overfitting and Underfitting, Introduction to Keras and TensorFlow and PyTorch.

Practice:
1. Installation and basic commands of scikit, TensorFlow and PyTorch
2. Implement Random Forest by using scikit, TensorFlow and PyTorch
3. Implement multilayer perceptron algorithm for MNIST Handwritten Digit Classification

UNIT-II

NEURAL NETWORKS : About Neural Network, Building Blocks of Neural Network, Introduction to Neural Networks, Feed-forward Networks, Deep Feed-forward Networks - Learning XOR, Gradient Based learning, Hidden Units, Back-propagation and other Differential Algorithms, Data Pre-processing for neural networks, Feature Engineering.

Practice:

1. Design a neural network for classifying movie reviews (Binary Classification) using IMDB dataset.
2. a) Design a neural Network for classifying news wires (Multi class classification) using Reuters dataset
   b) Design a neural network for predicting house prices using Boston Housing Price dataset.

UNIT-III

OPTIMIZATION TECHNIQUES: Optimizers – SGD (Stochastic Gradient Descent), Mini-batch Gradient Descent, MS prop (Root Mean Square Propagation),Adagrad (Adaptive Gradient Algorithm), Adadelta (Adaptive Delta), Adam (Adaptive Moment Estimation), Adamax (Adaptive Moment Estimation with Infinity Norm).

Activation Functions : sigmoid, tanh, ReLU, leaky ReLU, Soft Max, Linear Loss Functions - cross-entropy loss, the mean-squared error, the Huber loss, the hinge loss ,L2 loss , mean absolute loss Regularization – L1, L2 and drop out . Learning Rate, Normalization

Practice:

1. Implement word embeddings for IMDB dataset.
2. Implement one hot encoding of words or characters.
3. Consider Patient Dataset. Apply linear classification technique(SVM) to identify the rate of heart patients. Also apply the optimization techniques.
4. Write a Python program to construct a Bayesian network considering medical data. Use this model to demonstrate the diagnosis of heart patients using standard Heart Disease Data Set

UNIT-IV

CONVOLUTIONAL NEURAL NETWORK : About CNN, Building a convolutional neural network, Input Layers, Convolution Layers, Pooling Layers, Dense Layers, Backpropagation Through the Convolutional Layer, Filters and Feature Maps, Backpropagation Through the Pooling Layers, Dropout Layers and Regularization, Batch Normalization

Pre trained CNN: LeNet, Alex Net, VGG16, ResNet.

Practice:

1. Build a Convolution Neural Network for MNIST Handwritten Digit Classification.
2. a) Use a pre-trained convolution neural network (VGG16) for image classification
   b) Build a Convolution Neural Network for simple image (dogs and Cats) Classification
3. Student Portfolio Creation using GitHub, Hugging Face & Kaggle

UNIT – V

RNN : Recurrent Neural Networks ,Sequence-to-Sequence Modelling – Embedding - Recurrent Neural Networks - Bidirectional RNNs, Analysing Variable Length Inputs – Tackling seq2seq Problem – Beam Search and Global Normalization – Recurrent Neural Networks (RNN)– Hidden States – Perplexity – Character-level Language Models –Modern RNNs: Gated Recurrent Units (GRU), Long Short-Term Memory (LSTM).

Practice:

1. Implement a Recurrent Neural Network for IMDB movie review classification problem.
2. Implement Boosting algorithms using scikit, Tensorflow and PyTorch

Additional Practice:

1. Assuming a set of documents that need to be classified, use the naïve Bayesian Classifier model to perform this task. Built-in Java classes/API can be used to write the program. Calculate the accuracy, precision, and recall for your data set.
2. Apply EM algorithm to cluster a Heart Disease Data Set. Use the same data set for clustering using k-Means algorithm. Compare the results of these two algorithms and comment on the quality of clustering. You can add Java/Python ML library classes/API in the program.

Cornerstone Projects :

PROJECT 1: Student Performance Prediction using Deep Neural Networks
Project Implementation, Documentation and Viva Voce
Implementation mainly focuses on:

- ○ Collecting and preprocessing student academic datasets
- ○ Handling missing values and feature normalization
- ○ Applying feature engineering techniques
- ○ Building a Deep Feed Forward Neural Network using TensorFlow/Keras
- ○ Training the model for student performance prediction
- ○ Applying activation functions such as ReLU and Softmax
- ○ Using optimizers like SGD and Adam for model training
- ○ Evaluating prediction accuracy using confusion matrix and classification report
- ○ Comparing different optimization techniques and activation functions

Documentation must contain:

- ○ Objective and problem definition
- ○ Dataset description and preprocessing methods
- ○ Neural network architecture diagram
- ○ Optimizer and activation function comparison
- ○ Training and validation performance graphs
- ○ Prediction results and accuracy analysis
- ○ Challenges faced during implementation
- ○ Future enhancements using ensemble learning

PROJECT 2: Heart Disease Prediction System using Deep Learning
Project Implementation, Documentation and Viva Voce
Implementation mainly focuses on:

- ○ Loading and preprocessing heart disease datasets
- ○ Performing normalization and feature selection
- ○ Building Deep Neural Network models for disease prediction
- ○ Applying dropout and regularization techniques to reduce overfitting
- ○ Comparing DNN performance with SVM classifier
- ○ Using TensorFlow/PyTorch for implementation
- ○ Evaluating the model using accuracy, precision, recall, and F1-score
- ○ Creating a simple prediction interface/dashboard
- ○ Analyzing loss convergence and optimizer performance

Documentation must contain:

- ○ Objective and healthcare problem statement
- ○ Dataset details and preprocessing techniques
- ○ DNN architecture and workflow diagram
- ○ Optimization and regularization techniques used
- ○ Model evaluation metrics and confusion matrix
- ○ Prediction screenshots and analysis
- ○ Challenges and limitations
- ○ Future scope and enhancements

PROJECT 3: Handwritten Digit Recognition using Convolutional Neural Networks
Project Implementation, Documentation and Viva Voce
Implementation mainly focuses on:

- ○ Loading and preprocessing the MNIST dataset
- ○ Performing image normalization and reshaping
- ○ Building a CNN model with convolution, pooling, and dense layers
- ○ Applying dropout and batch normalization techniques
- ○ Training and validating the CNN model
- ○ Visualizing feature maps and convolution outputs
- ○ Predicting handwritten digits from custom images
- ○ Comparing CNN performance with traditional neural networks
- ○ Evaluating accuracy and loss convergence

Documentation must contain:

- ○ Objective and problem definition
- ○ Dataset description and preprocessing steps
- ○ CNN architecture diagram
- ○ Training parameters and optimizer details
- ○ Accuracy and loss analysis
- ○ Sample prediction outputs
- ○ Feature visualization results
- ○ Future enhancements using pre-trained CNN models

PROJECT 4: Animal Image Classification using Transfer Learning
Project Implementation, Documentation and Viva Voce
Implementation mainly focuses on:

- ○ Collecting and preprocessing animal image datasets
- ○ Performing image augmentation and normalization
- ○ Using pre-trained CNN models such as VGG16 or ResNet
- ○ Applying transfer learning for image classification
- ○ Fine-tuning dense layers for improved performance
- ○ Training and validating the classification model

- ○ Comparing transfer learning with custom CNN models
- ○ Predicting unseen animal images
- ○ Generating classification reports and confusion matrix

Documentation must contain:

- ○ Objective and application overview
- ○ Dataset preparation and augmentation methods
- ○ Transfer learning workflow diagram
- ○ Pre-trained model architecture explanation
- ○ Model evaluation and performance analysis
- ○ Accuracy comparison graphs
- ○ Prediction screenshots and outputs
- ○ Future enhancements using advanced CNN architectures

PROJECT 5: Movie Review Sentiment Analysis using LSTM

Project Implementation, Documentation and Viva Voce

Implementation mainly focuses on:

- ○ Loading and preprocessing IMDB movie review dataset
- ○ Performing text cleaning and tokenization
- ○ Applying one-hot encoding and word embeddings
- ○ Building an LSTM-based sentiment analysis model
- ○ Training the model for binary text classification
- ○ Using sequence padding and embedding layers
- ○ Evaluating model accuracy and loss convergence
- ○ Predicting sentiment for custom user reviews
- ○ Comparing RNN and LSTM performance

Documentation must contain:

- ○ Objective and problem statement
- ○ Dataset and vocabulary analysis
- ○ Text preprocessing methodology
- ○ LSTM architecture and sequence flow diagram
- ○ Training parameters and optimizer details
- ○ Accuracy/loss convergence graphs
- ○ Sample prediction outputs
- ○ Future scope using Bidirectional LSTM or Transformers

PROJECT 6: News Category Classification using RNN and Word Embeddings

Project Implementation, Documentation and Viva Voce

Implementation mainly focuses on:

- ○ Collecting and preprocessing news article datasets
- ○ Performing tokenization and sequence generation
- ○ Implementing one-hot encoding and word embeddings
- ○ Building RNN/LSTM models for multi-class text classification
- ○ Training the model to classify news into categories
- ○ Applying dropout and regularization techniques
- ○ Evaluating model performance using precision, recall, and F1-score
- ○ Visualizing training accuracy and loss graphs
- ○ Comparing Simple RNN and LSTM model performance

Documentation must contain:

- ○ Objective and problem definition
- ○ Dataset description and preprocessing techniques

○ RNN/LSTM architecture workflow
○ Embedding and sequence modeling explanation
○ Training configuration and evaluation metrics
○ Classification results and confusion matrix
○ Challenges in sequence modeling
○ Future enhancements using attention mechanisms

Text Books:

1 Deep Learning, Ian Goodfellow, Yoshua Bengio, Aaron Courville, MIT Press, ISBN: 9780262035613
2 Deep learning: A practitioner's approach, Josh Patterson and Adam Gibson, O'Reilly Media, First Edition, ISBN: 978-1491914250

Reference Books:

1 Fundamentals of Deep Learning, Designing next generation machine intelligence algorithms, Nikhil Buduma, O'Reilly, Shroff Publishers, ISBN: 978-9352135608
2 Deep learning Cookbook, Practical recipes to get started Quickly, Douwe Osinga, O'Reilly, Shroff Publishers, ISBN: 978-9352137572

Web Links:

1 https://onlinecourses.nptel.ac.in/noc20_cs62/preview
2 https://keras.io/datasets/

Note:

- The cornerstone project team size shall be four (4) students.
- Students may select any one of the above prescribed projects or a project of their own choice with the prior permission of the Course instructor.
- The evaluation of the Cornerstone Project shall be based on the project implementation, oral presentation, 10-minute video presentation, report and viva voce.
- The video presentation should consist of the working procedure of the project along with contribution of each student for a minimum of 2 minutes.
- Finally, the Source code of the cornerstone Project has to be pushed into the Students GitHub repository (if applicable).

Natural Language Processing & Prompt Engineering

	L	T	P	C
Course Code: 2501AI20	**2**	**0**	**1**	**3**

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

CO1: Explain Text Normalization and utilize Ngram language models for various NLP tasks.

CO2: Explain Text preprocessing and stemming algorithms.
CO3: Apply similarity measures and word embedding models
CO4: Summarize morphology and NLP applications
CO5: Explain prompt engineering and its applications.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	2	2	1	-	1	-	-	1	1	-	2
CO2	2	2	2	-	2	-	-	1	1	-	2
CO3	2	2	3	-	1	-	-	1	1	-	2
CO4	2	2	3	-	1	-	-	1	1	-	1
CO5	2	2	3	-	1	-	-	1	1	-	1

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1	-	1
CO2	-	2
CO3	-	2
CO4	-	2
CO5	-	2

UNIT-I
Introduction: Introduction to Natural Language Processing Overview of NLP Introduction to Levels of NLP –Regular Expressions, Text Normalization, Edit Distance Language Modelling: Ngram Language Models.

Practice:
1. Word Analysis Learn about morphological features of a word by analysing it
2. Word Generation generate word forms from root and suffix information

UNIT-II
Text Processing: Tokenization, Stemming, Text Preprocessing & Feature Representation Introduction to Corpora, Sentence Segmentation, Stemming: Porter Stemmer, Bag of words and Vector Space Model, Topic Modelling, Ngram Language Model, Smoothing,

Practice:
1. Implement Porter Stemming Algorithm
2. NGrams learn to calculate bigrams from a given corpus and calculate probability of a sentence

UNIT-III
Vector Semantics and Embeddings: Lexical Semantics, Vector, Words and Vectors, Cosine for measuring similarity, TFIDF: Weighing terms in the vector, Pointwise Mutual Information (PMI) , Applications of the tfidf or PPMI vector models , . Word Embeddings: Word2Vec, Glove and Fasttext, Visualizing Embeddings, Semantic properties of embeddings , Bias and Embeddings ,

Evaluating VectorModels .

Practice:

1. Apply term frequency"(TFIDF) to Measures the importance of a word to a specific document
2. Implement word embedding using word2vec

UNIT-IV

Morhpology: Sequence Labelling for Parts of Speech and Named Entities, PartofSpeech Tagging, Named Entities and Named Entity Tagging , HMM PartofSpeech Tagging , Conditional Random Fields (CRFs), Evaluation of Named Entity Recognition Applications of NLP Machine Translation Encoder & Decoder Model, Attention Models, Question Answering Knowledge based Q&A Chatbots & Dialogue Systems, Automatic Speech Recognition and Text to Speech.

Practice:

1. Morphology understanding the morphology of a word by the use of Add Delete table.
2. POS Tagging – calculate emission and transition matrix which will be helpful for tagging Parts of Speech using Hidden Markov Model.

UNIT – V

Introduction to Prompt Engineering : What is prompt engineering and why it matters
Prompt types: explicit, implicit, and creative prompts, Best Practices for Crafting Effective Prompts.
Practical Applications of Prompt Engineering : Improving NLP Tasks with Custom Prompts, Enhancing Creativity and Diversity in AI Generated Content, Addressing AI Ethics and Bias through Thoughtful Prompt Engineering, Personalization and Adaptability in AI Generated Content.
Future Directions and Emerging Trends in Prompt Engineering : Leveraging Advanced AI Models and Techniques, The Convergence of Human and AI Creativity, The Role of Prompt Engineering in the AI Driven Economy

Practice:

1. Chunking understand the concept of chunking and get familiar with the basic chunk tagset.
2. Building Chunker the importance of selecting proper features for training a model and size of training corpus in learning how to do cunking.

Additional Practice:

1. Text normalization is a crucial process in natural language processing (NLP) that involves cleaning and preparing text data to ensure consistency and usability across different NLP tasks. By standardizing text, we reduce data variability, making it easier for NLP models to identify patterns and generate accurate predictions.
   Apply text normalization to the following text
   Jaron Lanier said: “It would be unfair to demand that people cease pirating files when those same people aren’t paid for their participation in very lucrative network schemes. Ordinary people are relentlessly spied on, and not compensated for information taken from them. While I’d like to see everyone eventually pay for music and the like, I’d not ask for it until there’s reciprocity.”
2. Demonstrate text smoothing using Laplace Smoothing (Additive Smoothing) for a simple bigram language model.
3. For the give documents data Interpret the documents that are closely related or similar in

content, apply TFIDF and Cosine Similarity
Doc1 = "The quick brown fox jumps over the lazy dog"
Doc2 = "Never jump over the lazy dog quickly"
Doc3 = "Bright blue fox jumps quickly"
Doc4 = "The fox is quick and the dog is lazy"

Capstone Project
Sentiment Analysis and Emotion Detection in Tweets or Product Reviews

Objective: **Analyze social media or review texts to detect sentiment (positive/negative) and optionally the emotion (joy, anger, etc.).**

Domain: **Social Media / E-commerce / Healthcare Communication**

Techniques & Concepts Used:

- Text Preprocessing and Tokenization (Unit II)
- Word Embeddings (Word2Vec/FastText) (Unit III)
- Sequence Labeling for emotion classes (Unit IV)
- Prompt Engineering (for custom sentiment questions) (Unit V)

Text Books:

1 Speech and Language Processing, Dan Jurafsky, James H. Martin Draft of 3rd Edition, Prentice Hall 2022, ISBN: 978-0130950697
2 Springer Handbook of Speech Processing, Jacob Benesty, M. M. Sondhi, Yiteng Huang Springer, ISBN: 978-3662533000

Reference Books:

1 Speech and Language Processing, Daniel Jurafsky and James H. Martin. 2020. 3rd Edition (draft), ISBN: 978-0135041963
2 Foundations of Statistical Natural Language Processing, Christopher, D. Manning and Hinrich Schütze, MIT Press, ISBN: 978-0262133609

Web Links:

1 https://www.nltk.org/data.html
2 https://www.analyticsvidhya.com/blog/2017/01/ultimateguidetounderstandimplement naturallanguageprocessingcodesinpython/
3 https://datahack.analyticsvidhya.com/contest/linguipediacodefestnaturallanguageprocessing1/?utm_1source=ultimateguidetounderstandimplementnaturallanguageprocessingcodesinpython&utm_medium=blog

Reinforcement Learning & Gen AI

	L	T	P	C
Course Code: 2501AI21	**2**	**0**	**1**	**3**

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

CO1: Outline the basic concepts of Reinforcement learning.
CO2: Identify the appropriate learning tasks for Reinforcement learning techniques.

CO3: Implement adversarial training techniques to train GAN models.

CO4: Analyse the applications of VAEs in various domains such as image generation, anomaly detection, and data imputation.

CO5: Evaluate the ethical considerations and societal impact of reinforcement learning and generative AI technologies.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	2	2	-	-	2	-	-	-	1	-	-
CO2	2	2	-	-	2	-	-	-	1	-	-
CO3	3	2	-	-	2	-	-	-	1	-	-
CO4	3	2	1	-	2	-	-	-	1	-	-
CO5	2	1	1	-	2	-	-	-	1	-	-

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1	2	-
CO2	2	1
CO3	3	1
CO4	2	-
CO5	3	1

UNIT-I

Basics of reinforcement learning (RL), RL components: agents, environments, rewards Markov Decision Processes (MDPs), Exploration vs. Exploitation dilemma
Basic algorithms: Qlearning, SARSA, MonteCarlo (MC) Learning, TemporalDifference (TD) Learning, TDLambda and Eligibility Traces.

Practice:

1. Installation of TensorFlow and implement the basic programs.
2. Installation of PYTORCH and implement the basic programs.

UNIT-II

Advanced Reinforcement Learning Techniques: Deep QNetworks (DQN), Policy Gradient methods, ActorCritic architectures, Advantage ActorCritic (A2C) and Proximal Policy Optimization (PPO), Continuous action spaces and algorithms like Deep Deterministic Policy Gradient (DDPG).

Practice:

1. Implement a simple grid world environment where an agent learns to navigate from a starting point to a goal while avoiding obstacles.
2. Train an agent to balance a pole on a cart using discrete actions such as pushing the cart left or right.

UNIT-III

Generative Adversarial Networks (GANs): Introduction to generative models, Basics of

GANs: generator, discriminator ,Training GANs: adversarial training ,Variants of GANs: Conditional GANs, Wasserstein GANs, etc, Applications of GANs in image generation, style transfer, and data augmentation.

Practice:

1. Model an agent learning to drive a car up a steep mountain by applying the correct amount of throttle and braking.
2. Train an agent to play the classic game of Pong, where it learns to control a paddle to hit a ball back to the opponent while preventing it from reaching its own side.

UNIT-IV

Variational Autoencoders (VAEs): Introduction to autoencoders , Variational inference and latent variable models ,Encoder and decoder architectures in VAEs , Training VAEs: maximizing evidence lower bound (ELBO) , Applications of VAEs in image generation, anomaly detection, and data imputation.

Practice:

1. Use deep reinforcement learning to train an agent to play various Atari games, such as Breakout, Space Invaders, or PacMan.
2. Experiment with different strategies for solving the multiarmed bandit problem, where an agent must decide which arm of a slot machine to pull to maximize cumulative reward.

UNIT – V

Advanced Topics in Reinforcement Learning and Generative AI: Model based RL and world models, multiagent reinforcement learning ,Transfer learning and meta learning in RL, Adversarial attacks and defences in generative models, Ethical considerations and societal impact of RL and generative AI, LangChain Framework

Practice:

1. Train an agent to play Flappy Bird, a sidescrolling game where the agent controls the flight of a bird through a series of pipes.
2. Develop an agent capable of playing TicTacToe optimally against a human opponent or another agent.

Additional Practice:

1. Train a robotic arm to perform various tasks, such as reaching a target location or manipulating objects, using reinforcement learning.
2. Develop a trading agent that learns to make profitable trades in financial markets by analysing historical data and adapting its trading strategy over time

Capstone Project

Smart Traffic Signal Control Using Q-Learning

Objective: Optimize traffic signal timings at intersections to reduce vehicle waiting

time using RL.
Domain: Smart Cities / Traffic Management
Techniques & Concepts Used:
*Q-Learning (Tabular RL)
*State and Reward Modeling
*Traffic Simulation Environment (e.g., SUMO or custom gridworld)
*Visualization of Learning Process
*Performance Metrics: Average wait time, throughput

Text Books:

1 Reinforcement Learning An Introduction, R. S. Sutton and A. G. Bart,. MIT Press, 2018, ISBN: 978-0262039246

Reference Books:

1 Algorithms for Reinforcement Learning, Szepesvári, Csaba, United States: Morgan &Claypool, ISBN: 978-1627050278
2 Markov Decision Processes, Discrete Stochastic Dynamic Programming, Puterman, Martin L., Germany, Wiley, ISBN: 978-0471727200
3 Artificial Intelligence, Dr. R. Sravanth Kumar,Dr. A. Mallikarjuna Reddy,Dr T. Prabhakara Rao and Dr. Gayatri Parasa Published By Pandit Publications-2024,ISBN:978-93-93769-74-9

Web Links:

1 https://onlinecourses.nptel.ac.in/noc20_cs74/preview
2 https://www.coursera.org/learn/fundamentalsofreinforcementlearning

Big Data Analytics

(Common to CSE, AIML & CSE(DS))

	L	T	P	C
Course Code: 2501AI04	**2**	**0**	**1**	**3**

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

CO1: Illustrate big data challenges in different domains

CO2: Use various techniques for mining data stream.
CO3: Demonstrate Building blocks of Hadoop
CO4: Make use of Pig and Hive to structure and work with big Data
CO5: Make use of Spark tool to work with big data.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	3	2	-	1	2	-	-	1	2	-	1
CO2	2	3	-	1	2	-	-	1	2	-	1
CO3	2	1	-	3	2	-	-	1	2	-	2
CO4	2	3	-	2	2	-	-	1	2	-	2
CO5	2	2	-	1	3	-	-	1	2	-	2

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1	-	2
CO2	-	2
CO3	-	2
CO4	-	2
CO5	-	2

UNIT-I

Introduction to big data: Introduction to Big Data Platform, Challenges of Conventional Systems, Intelligent data analysis, Nature of Data, Analytic Processes and Tools, Analysis vs Reporting.

Practice:

1. Hadoop File System Operations: Implement the following file management tasks in Hadoop:
   a.) Adding files and directories b.) Retrieving files c.) Deleting files

UNIT-II

Stream Processing:

Mining data streams: Introduction to Streams Concepts, Stream Data Model and Architecture, Stream Computing, Sampling Data in a Stream, Filtering Streams, Counting Distinct Elements in a Stream, Estimating Moments, Counting Oneness in a Window, Decaying Window, Real time Analytics Platform (RTAP) Applications.

Practice:

1. Loading DataSet in to HDFS for Spark Analysis.

UNIT-III

Introduction to Hadoop: History of Hadoop, the Hadoop Distributed File System, Components of Hadoop Analyzing the Data with Hadoop, Scaling Out, Hadoop Streaming, Design of HDFS, Java interfaces to HDFS Basics, Developing a Map Reduce Application,

How Map Reduce Works, Anatomy of a Map Reduce Job run, Failures, Job Scheduling, Shuffle and Sort, Task execution, Map Reduce Types and Formats, Map Reduce Features Hadoop environment.

Practice:

1. Map Reduce Word Count Hadoop Run a basic Word Count Map Reduce program to understand Map Reduce Paradigm

UNIT-IV

Pig: Hadoop Programming Made Easier: Admiring the Pig Architecture, Going with the Pig Latin Application Flow, Working through the ABCs of Pig Latin, Checking out the Pig Script Interfaces, Scripting with Pig Latin. Working with Hive Data Types, Creating and Managing Databases and Tables, Seeing How the Hive Data Manipulation Language Works with examples, Querying and Analyzing Data.

Practice:

1. Hive Queries a) Install Hive Framework b) Implement Hive to create, alter, and drop databases, tables
2. Hive Queries Implement hive queries and joins to perform display and retrieve the data

UNIT – V

Spark: Installing Spark, Spark applications, Jobs, stages and Tasks, Resilient Distributed data sets, Shared Variables, Anatomy of a Spark job run.

Practice:

1. Demonstrate Spark SQL on Hive
   a. Create a SQLContext object and load the Parquet file into DataFrame
   b. Load the Dataframe into Hive table.
   c. Verify the created Hive table in Hive environment
   d. Execute Spark SQL query

Text Books:

1 Hadoop: The Definitive Guide, Tom White, O'reilly, 4th Edition, ISBN: 978-1491901632

2 Hadoop for Dummies, Dirk deRoos, Paul C.Zikopoulos, Roman B.Melnyk, Bruce Brown, Rafael Coss, John Wiley & Sons, ISBN: 978-1118727128

Reference Books:

1 Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with

Advanced Analytics, Bill Franks, John Wiley& sons, ISBN: 978-1118224912
2 Harness the Power of Big Data:The IBM Big Data Platform, Paul Zikopoulos, DirkdeRoos, Krishnan Parasuraman, Thomas Deutsch, James Giles, David Corrigan, Tata McGraw Hill Publications, ISBN: 9780071808187
3 Big Data Science & Analytics: A Hands On Approach , Arshdeep Bahga and Vijay Madisetti, VPT, ISBN: 978-0996025577
4 Learning Spark: Lightning Fast Big Data Analysis Paperback, Holden Karau, ISBN: 978-1449358624

Web Links:

1 Hadoop: http://hadoop.apache.org/
2 Hive: https://cwiki.apache.org/confluence/display/Hive/Home
3 https://nptel.ac.in/courses/106106142
4 https://www.cloudera.com/servicesandsupport/tutorials.html
5 https://www.databricks.com/spark/gettingstartedwithapachespark

Multidisciplinary Courses (MDC)

Course Code	Course Name	Level	L	T	P	C	CIE	SEE	Total	Pre-requisite
2501EE01	Basic Electrical & Electronics Engineering	**FC**	2		2	4	50	50	100	-
2501EC95	Digital Logic Design	**FC**	2	1		3	50	50	100	-
2501MB01	Engineering Economics &	**FC**	2			2	50	50	100	-

	Management									
Total			**Total**	**6**	**1**	**2**	**9**			

Basic Electrical & Electronics Engineering

(Common to CE, ME, ECE, CSE, IT, AIML, CSE(DS), PT & Min.E)

	L	**T**	**P**	**C**
Course Code: 2501EE01	**2**	**0**	**2**	**4**

Course Outcomes:

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

CO1: Analyze the concepts associated to AC and DC circuits.

CO2: Explain the operating principles of motors, generators and measuring instruments.
CO3: Analyze the Different Energy Resources and Equipment Safety Measures.
CO4: Explain the concept and the applications of semiconductor Diodes.
CO5: Analyze the Basic Electronic Circuits and interpret numeric information in different code formats.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	2	3	1	-	1	-	-	-	2	2	-
CO2	2	3	1	-	-	-	-	-	2	2	-
CO3	3	2	1	-	-	-	-	-	2	2	-
CO4	3	2	1	-	-	-	-	-	2	2	-
CO5	3	2	1	-	-	-	-	-	2	2	-

UNIT-I

DC & AC Circuits:
DC circuits: Ohm's and Kirchhoff's laws, analysis of series, parallel and series-parallel circuits excited by independent voltage sources for R, L, C parameters, current division, voltage division
AC circuits: Generation of sinusoidal voltage, frequency of generated voltage, average value, RMS value, form, and peak factors. Real power, reactive power, apparent power, and Power factor.

Practice:
1. Verification of ohms law
2. Verification of KCL and KVL
3. Verification of KCL, KVL and ohm's law using simulation

UNIT-II

Machines and Measuring Instruments: Principles and operation of DC machines, Transformers – Synchronous Machines - three Phase and single phase induction motors - Moving coil and moving iron instruments, Wheatstone bridge and Megger

Practice:
1. Magnetisation Characteristics of DC shunt generator
2. Measurement of Power and Power factor using Single-phase wattmeter
3. Measurement of Resistance using Wheat stone bridge
4. Measurement of Earth Resistance using Megger.

UNIT-III

Energy Resources, Electricity Bill & Safety Measures: Conventional and non-conventional energy resources; Layout and operation of various Power Generation systems: Hydel, Thermal, Solar and Wind power generation. Calculation of electricity bill for domestic appliances. Working principle of Fuse and Miniature circuit breaker (MCB). Electric Shock, Earthing and its

types, Safety Precautions to avoid shock.

Practice:

1. Calculation of Electrical Energy for Domestic Premises

UNIT-IV

Semiconductor Devices: Intrinsic semiconductors – Extrinsic semiconductors - P type and N type - P-N junction characteristics of P N Junction Diode — Zener Effect — Zener Diode and its Characteristics. working of simple zener voltage regulator and amplifier- Bipolar Junction Transistor — CB, CE, CC Configurations and Characteristics.

Practice:

1. Sketch the V-I characteristics of PN Junction diode A) Forward bias B) Reverse bias.
2. Sketch the V-I characteristics of Zener Diode and its application as voltage Regulator
3. Plot Input & Output characteristics of BJT in CE and CB configurations.
4. Obtain Frequency response of CE amplifier

UNIT – V

BASIC ELECTRONIC CIRCUITS: Block diagram description of a dc power supply, working of a half and full wave, bridge rectifier, filters.

DIGITAL ELECTRONICS: Overview of Number Systems, Logic gates including Universal Gates, BCD codes, Excess-3 code, Gray code, Hamming code. Truth Tables and Functionality of Logic Gates – NOT, OR, AND, NOR, NAND, XOR and XNOR. Simple combinational circuits–Half and Full Adders

Practice:

1. Implementation of half wave and full wave rectifiers.
2. Design Half Adder and Full Adder circuits.
3. Verification of truth table for Logic gates using ICs

Text Books:

1 Basic Electrical and Electronics Engineering, Salivahanan S, Tata McGraw Hill Education (India) Private Limited, New Delhi, ISBN: 9789389691801

2 Principles of Electrical Engineering, V. K. Mehta, R. Mehta, S. Chand & Company Ltd., New Delhi, ISBN-13: 9788121930888

3 Digital Fundamentals, Thomas Floyd, Prentice Hall, 10th Edition, ISBN: 9780132737968

Reference Books:

1 Electronic Devices & Circuit Theory, Robert L. Boylestad and Louis Nashelsky, Pearson, 11/e, ISBN: 9780135026496
2 Power System Engineering, P.V. Gupta, M.L. Soni, U.S. Bhatnagar and A. Chakrabarti, Dhanpat Rai & Co, ISBN: 9788177000207

Web Links:

1 https://nptel.ac.in/courses/108/101/108101091/ (NPTEL Video by Dr.Mahesh B. Patil from IIT Bombay)
2 https://nptel.ac.in/courses/117/106/117106108/ (NPTEL Video by Prof. Nagendra Krishnapura from IIT Madras)

Digital Logic Design

(Common to CSE, IT, AIML & CSE(DS))

	L	T	P	C
Course Code: 2501EC95	**2**	**1**	**0**	**3**

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

CO1: Understand various number systems, perform conversions between them, and apply binary

codes in digital representations. (BTL: 2)

CO2: Understand basic Boolean algebra rules and simplify Boolean expressions using Karnaugh maps. (BTL: 2)

CO3: Design and implement basic combinational logic circuits such as adders, decoders, and multiplexers. (BTL: 3)

CO4: Apply the concepts of flip-flops to implement basic sequential circuits like registers, and counters. (BTL: 3)

CO5: Develop logic circuits using PLDs such as PROM, PAL, and PLA by utilizing programming tables. (BTL: 3)

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	3	2	-	-	-	-	-	2	1	-	1
CO2	2	2	-	-	-	-	-	2	1	-	1
CO3	2	2	3	1	-	-	-	2	1	-	1
CO4	2	2	3	1	-	-	-	2	1	-	1
CO5	2	2	3	1	-	-	-	2	1	-	1

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1	-	2
CO2	-	2
CO3	-	3
CO4	-	3
CO5	-	3

UNIT-I

Digital Systems and Binary Numbers

Digital Systems, Binary Numbers, Number based Conversions, Octal & Hexadecimal Numbers, Complements, Signed binary Numbers, Arithmetic addition and subtraction.

Binary Codes: BCD, 2421, 8421, Excess-3, Gray codes

UNIT-II

Concept of Boolean algebra and Gate Level Minimization

Basic Definitions, Axioms and laws, Basic Theorems, Boolean Functions Simplification .

Digital logic gates, The Map Method –K-Maps(up to 4 variable). Product of Sums Simplification, Sum of Products Simplification, Don't Care Conditions.

UNIT-III

Combinational Logic

Full adders, Full subtractor, Ripple carry adder, 4-bit Binary adder/subtractor circuit, Decoders(up to 3*8), Encoders(up to 8*3), Multiplexers(up to 8*1), De-Multiplexers(up to 1*8), One bit comparator.

UNIT-IV

Sequential Circuits
Introduction to Sequential Circuits, Latches, RS- Latch Using NAND and NOR Gates, Flipflops(RS, JK, D and T)Truth Tables.
Registers and Counters
Registers, Shift Registers, Ripple Counters – Binary Ripple counter, 3 Bit synchronous counter, Ring Counter.

UNIT – V

Programmable Logic Devices: Introduction to PLDs: PROM, PAL, PLA-Basics structures, realization of Boolean function with PLDs, comparison, merits & demerits of PROM, PAL, PLA.

Text Books:

1 Switching and Finite Automata Theory, ZviKohavi& Niraj K.Jha, 3rdEdition, ISBN: 978-0521857482.
2 Digital Design, Morris Mano, Pearson, 3rd Edition, ISBN: 978-8178085555.

Reference Books:

1 Modern Digital Electronics, RP Jain, Tata Mc Graw Hill, 5thEdition, 2022, ISBN: 978-9355321770.
2 Fundamentals of Logic Design, Charles H. Roth Jr., Jaico Publishers, ISBN: 978-0534378042.

Web Links:

1 http://nptel.ac.in/courses/117/106/117106086/ (By Prof. Goutam Saha, Electronics & Electrical Communication Engineering Dept, IIT Kharagpur)
2 www.nptelvideos.in/2012/12/digital-circuits-and-systems.html(By Prof. Santanu Chattopadhyay, Electronics & Electrical Communication Engineering Dept, IIT Kharagpur).
3 https://www.smartzworld.com/notes/switching-theory-and-logic-design-stld/.

Engineering Economics & Management

(Common to CE, EEE, ME, CSE, IT, AIML & CSE(DS))

	L	T	P	C
Course Code: 2501MB01	**2**	**0**	**0**	**2**

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

CO1: Explain the Business Economic concepts, law of demand and forecasting methods.

CO2: Identify the production, cost behaviour for managerial decision making with Break-Even Point (BEP).

CO3: Make use of financial accounting and capital budgeting techniques for decision making.

CO4: Summarize management and motivational theories to renovate the practice of Management.

CO5: Illustrate the functional management and project management using PERT and CPM.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	-	-	-	-	-	-	-	2	2	-	-
CO2	1	-	-	-	-	-	-	-	-	2	-
CO3	-	-	-	-	-	-	-	-	-	3	-
CO4	1	-	-	-	-	-	-	1	1	2	-
CO5	-	-	-	-	-	-	-	-	-	3	-

UNIT-I

Introduction to Managerial Economics and demand Analysis: Definition of Managerial Economics –Scope of Managerial Economics- Concept of Demand, Types of Demand, Determinants of Demand- Law of Demand and its limitations- Elasticity of Demand, Types- Demand forecasting and its Methods.

UNIT-II

Production and Cost Analyses: Concept of Production function Law of Variable Proportions-Isoquants and Isocosts -Producer Equillibrium-, cost concepts: opportunity costs, explicit and implicit costs- Fixed costs, Variable Costs – Cost –Volume-Profit Analysis-Determination of Breakeven point (simple problems).

UNIT-III

Introduction to Markets and Financial Accounting: Market Structures-Classification of markets,.Introduction to Financial Accounting , Concepts and conventions, Accounting cycle, Journal entries and Ledger (Simple Problems), Methods of capital budgeting (Simple Problems).

UNIT-IV

Operations Management : Concept nature and importance of Management, Generic Functions of Management, Theories of Motivation, Plant location and layout, Principles of organization,SWOT analysis.
Material Management: Need for Inventory control, EOQ, ABC analysis

UNIT – V

Functional Management And Project Management: Concept of HRM , HRD and

PMIR, Functions of HR Manager , Job Evaluation and Merit Rating , Marketing Management, Functions of Marketing , Channels of distributions - Development of Network , Difference between PERT and CPM, Finding Critical Path (Simple Problems)

Text Books:

1 Managerial Economics and Financial Analysis, A. R. Aryasri , McGraw Hill Education, ISBN: 978-0070078031
2 Managerial Economics and Financial Analysis', N. Appa Rao, P. Vijay Kumar, Cengage Publications, New Delhi, ISBN: 978-8131515952
3 Management Science by Aryasri; Publisher: Tata McGraw Hill, 2009, ISBN: 978-0070090279
4 Management by James Arthur, Finch Stoner, R. Edward Freeman, and Daniel R. Gilbert 6th Ed; Publisher: Pearson Education/Prentice Hall, ISBN: 978-0131087477

Reference Books:

1 Principles of Marketing: A South Asian Perspective by Kotler Philip, Gary Armstrong, Prafulla Y. Agnihotri, and Eshanul Haque, 13th Edition, Publisher: Pearson Education/ Prentice Hall of India, ISBN: 9788131731017
2 A Handbook of Human Resource Management Practice by Michael Armstrong, 2010; Publisher: Kogan Page Publishers, ISBN: 978-1789661033

Web Links:

1 www.managementstudyguide.com
2 www.citehr.com
3 www.nptel.ac.in/courses/122106032
4 www.btechguru.com/courses--nptel--basic-course

Ability Enhancement Courses (AEC)

Course Code	Course Name	Level	L	T	P	C	CIE	SEE	Total	Pre-requisite
2501EN01	Essential Cognitive Skills for Engineers	**FC**			1	1	100	-	100	-
2501EN02	Advanced Cognitive Skills for Engineers	**FC**			1	1	100	-	100	ECSE-I
2501UC07	Design Thinking using AI	**FC**			1	1	100	-	100	-

2501UC08	Universal Human Values	**FC**	2			2	100	-	100	-
2501UC09	Technical Paper Publication	AC			2	2	100	-	100	-
2501AI35	Student Activity Based Learning	AC				2				
Total			**2**		**5**	**9**				

Essential Cognitive Skills for Engineers

(Common to CE, EEE, ME, ECE, CSE, IT, AIML, CSE(DS), PT & Min.E)

	L	**T**	**P**	**C**
Course Code: 2501EN01	**0**	**0**	**1**	**1**

Course Outcomes:

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

CO1: Recognize the different aspects of the English language proficiency with emphasis on LSRW skills.
CO2: Apply communication skills through various language learning activities.
CO3: Analyze the English speech sounds, stress, intonation and syllable division for better listening and speaking comprehension.
CO4: Enable them to learn and apply fundamentals of English grammar concepts for improved language
CO5: Make use of various types of vocabulary in different academic and professional careers

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	-	-	-	-	-	-	-	-	3	-	-
CO2	-	-	-	-	-	-	-	-	3	-	-
CO3	-	-	-	-	-	-	-	-	3	-	-
CO4	-	-	-	-	-	-	-	-	3	-	-
CO5	-	-	-	-	-	-	-	-	3	-	-

Practice:

UNIT – I **Outstanding people A/B/C/D**

	Conversation about Jocelyn Bell- Burnell, Podcast: The 30-day challenge, Starting a new job, Conversation about technology
Speaking	Discussing inspiring people, Asking and answering questions about challenges, Explaining a process; Checking understanding, Discussing technology.
Reading	Articles: *Protector of the sea and The woman who reinvented children's TV,* Interviews*: 30-day challenge,* Article: *Tech free!*
Writing	Article Organizing an article
Grammar	Review of Tenses, Questions
Vocabulary	Character adjectives, trying and succeeding
Pronunciation	The letter e; Word stress, Rapid speech

UNIT – II **Survival A/B/C/D**

Listening	Conversation about a survival situation, Interview: The Tiger, Cooking for a friend, Talking about getting lost
Speaking	Telling a survival story, Giving advice; Asking questions, Giving compliments and responding, Discussing the natural environment
Reading	Article: Lost at sea, Leaflet: *How to survive…an animal attack*, Leaflet: *Be wise and survive*
Writing	Guidelines, organizing guidelines in a leaflet
Grammar	Narrative tenses, Future time clauses and conditionals
Vocabulary	Expressions with *get,* Animals and the environment
Pronunciation	Sound and Spelling: g, Intonation in question tags

UNIT – III	**Talent A/B/C/D**
Listening	Conversation: learning experiences, Radio Programme: The sports gene, Making wedding plans, Interviews about sport
Speaking	Talking about something you have put a lot of effort into, Discussing sport and ways to improve performance, planning a party, Talking about popular sports
Reading	Text about learning; *Learning to learn*, Article: *Born to be the best*; Three articles about athletes, Article: *Fitness: Seattle snapshot*
Writing	Article describing data
Grammar	Multi-word verbs, Present perfect and present perfect continuous
Vocabulary	Ability and achievement, word connected with sport
Pronunciation	Word stress, sound and spelling consonant sounds
UNIT – IV	**Life Lessons A/B/C/D**
Listening	Interview: Psychology of money; Two monologues: Life-changing events, Two monologues; training for a job, Presenting photos, Three monologues; living in different places
Speaking	Talking about how your life has changes, Discuss experiences of training and rules, Describing photos: Expressing careful disagreement, Discussing living in a different country
Reading	Two texts about life-changing events that helped people become rich, Article: *Training for the emergency frontline,* Advert for being an international student 'buddy'
Writing	Job application, Giving a positive impression
Grammar	*Used to* and *would*
Vocabulary	Cause and result, Talking about difficulty
Pronunciation	Sound and spelling: *u*
UNIT – V	**Chance A/B/C/D**
Listening	Monologue: What are your chances?, Conversation: Talking about work, Money problems, News reports: environmental problems
Speaking	Discussing possible future events, Role Play: job interview, Explaining and responding to an idea for a café, Giving opinions on environmental problems
Reading	Quiz: *Are you an optimist or a pessimist?; Article: Why we think we're going to have a long and happy life*, Quiz: *The unknown continent*; Article: *Cooking in Antarctica*, Essay about protecting the environment
Writing	For and against essay, Arguing for and against an idea
Grammar	Future probability, Future perfect and future continuous
Vocabulary	Adjectives describing attitude, The natural world
Pronunciation	Sound and spelling : *th,* Intonation groups

Text Book:

1. Cambridge Empower – Second Edition B2 Level - Adrian Doff, Craig Thaine, Herbert Puchta, Jeff Stranks, Peter Lewis – Jones. ISBN: 978-1107468726

Suggested Software:

- Cambridge Empower
- Soft X (K-Van Solutions)

Reference Books:

1 Effective Technical Communication, M Ashraf Rizvi, ISBN: 9352605780
2 Raymond Murphy: English Grammar in Use, Cambridge University Press. Fifth Edition, ISBN: 978-3125354241
3 J. Sethi & P.V. Dhamija. A Course in Phonetics and Spoken English, (2nd Ed), Kindle, 2013, ISBN: 9788120314955

Web Links:

1 www.cambridgeone.org/login
2 https://www.britishcouncil.in/english/online
3 www.englishmedialab.com

Advanced Cognitive Skills for Engineers

(Common to CE, EEE, ME, ECE, CSE, IT, AIML, CSE(DS), PT & Min.E)

	L	T	P	C
Course Code: 2501EN02	**0**	**0**	**1**	**1**

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

CO1: Recognize the basics of communication and summarize formal and informal language expressions in all aspects.

CO2: Establish and maintain interpersonal relationships and transmit the message through different language activities.

CO3: Use language effectively to prepare and demonstrate proficiency in facing various types of interviews.

CO4: Demonstrate and exhibit professionalism in participating in various public speaking activities like debates, group discussions and presentation skills.

CO5: Identify the basic elements of writing and apply the fundamentals to compose e-mails catering to different professional needs.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	-	-	-	-	-	-	-	-	3	-	-
CO2	-	-	-	-	-	-	-	-	3	-	-
CO3	-	-	-	-	-	-	-	-	3	-	-
CO4	-	-	-	-	-	-	-	-	3	-	-
CO5	-	-	-	-	-	-	-	-	3	-	-

List of the Topics

UNIT – I **Around the globe A/B/C/D**

Listening	Two monologues about sightseeing tours, Interview: disappearing languages, Asking for a favour, conversation: a trip to the Grand Canyon
Speaking	Comparing different tourist destinations, Agreeing and disagreeing, Asking for a favour, Discussing local tourist destinations.
Reading	Website bout four tourist destinations; Website: *Where to go?*
Writing	Travel blog, Using descriptive language
Grammar	Infinitives and *–ing* forms
Vocabulary	Travel and tourism, Describing changes
Pronunciation	Consonant clusters, Consonant sounds

UNIT – II **City living A/B/C/D**

Listening	Interview: 'Smart' cities: Two monologues talking about 'smart' cities, Two monologues: house renovations, Flat hunting, Interviews about a new shopping centre
Speaking	Discussing good and bad points about a city
Reading	Article: *Quick-slow down!*, Article: *Who puts the 'real' in reality* TV? Email: Complaining about an important issue.
Writing	Email of complaint, Using formal language
Grammar	*Too/enough; so/such*, Causative *have/get*
Vocabulary	Describing life in cities, Film and TV; Houses
Pronunciation	Sound and spelling: o, Stress in compound nouns

UNIT – III	**Dilemmas A/B/C/D**
Listening	Radio programme: person finance, Three monologues about honesty, Going to the bank, Conversation about a TV programme
Speaking	Giving opinions on financial matters, Discussing moral dilemmas, Talking about hopes and worries Discussing programmes about crime
Reading	Article: *Is it time to give up on cash?*, Newspaper article: *The honesty experiment*, Review: *Crime with a smile*
Writing	Review, Organising a review
Grammar	First and second conditionals, Third conditional; *should have+past participle*
Vocabulary	Money and finance, Crime
Pronunciation	Stressed and unstressed words; Sound and spelling: *l,* Word groups
UNIT – IV	**Discoveries A/B/C/D**
Listening	Conversation about inventions, Conversation about an email hoax, Finding the perfect flat, Four monologues about alternative medicine
Speaking	Talking about inventions, Describing a hoax or a scam or a case of fraud, Giving and receiving surprises
Reading	Article: *Too good to be true?*, Article: *The rise and fall of Barry Minkow,* Essay: *The Value of alternative medicine*
Writing	Opinion essay, Presenting a series of arguments
Grammar	Relative clauses, Reported speech; Reporting verbs
Vocabulary	Health, Verbs describing thought and knowledge
Pronunciation	Sound and spelling : *ui*, Linking and intrusion
UNIT – V	**Possibilities A/B/C/D**
Listening	Interview about Dan Cooper, Two monologues: pursuing a dream, Celebrating good news, conversation about goals
Speaking	Telling stories about coincidences, Describing and comparing brave or amazing people, Telling an important piece of news, Talking about performing
Reading	Story: The man who disappeared; Blog: *The Wreck of the Titan*, Article: *Dream to help*, Story: Rosa's diary: *The ultimate goal*
Writing	Story, Making a story interesting
Grammar	Past modals of deduction, Wishes and regrets
Vocabulary	Adjectives with prefixes, Verbs of effort
Pronunciation	Word stress, Linking, Consonant clusters

Text Book:

1. Cambridge Empower – Second Edition B2 Level - Adrian Doff, Craig Thaine, Herbert Puchta, Jeff Stranks, Peter Lewis – Jones. ISBN: 978-1107468726

Suggested Software:

- Cambridge Empower
- Soft X (K-Van Solutions)

Reference Books:

1 Technical Communication. Raman Meenakshi, Sangeeta-Sharma. Oxford University Press.2018, ISBN: 97893549772256
2 Practical English Usage -Michael Swan, ISBN: 9780194202466
3 English Conversation Practice, Taylor Grant: Tata McGraw-Hill Education India, ISBN: 978-0070996038

Web links:

1 https://www.cambridgeone.org/login
2 https://www.coursera.org/
3 https://www.skillshare.com/
4 https://www.mindtools.com/cawh8bu/communication-tools

Design Thinking using AI

(Common to CE, EEE, ME, ECE, CSE, IT, AIML, CSE(DS), PT & Min.E)

	L	T	P	C
Course Code: 2501UC07	**0**	**0**	**1**	**1**

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

CO1: Appreciate various design process procedure
CO2: Generate and develop design ideas through different technique
CO3: Identify the significance of reverse Engineering to Understand products
CO4: Draw technical drawing for design ideas
CO5: Illustrate design teams to create feasible and user-focused solutions to complex problems in design.

Mapping of Course Outcomes with Program Outcomes:

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	-	-	-	-	-	-	-	2	2	-	-
CO2	-	-	-	-	-	-	-	-	-	2	-
CO3	-	-	-	-	-	-	-	-	-	3	-
CO4	-	-	-	-	-	-	-	1	1	2	-
CO5	-	-	-	-	-	-	-	-	-	3	-

UNIT-I

Process of Design Understanding Design thinking: Shared model in team-based design – Theory and practice in Design thinking – Explore presentation signers across globe – MVP or Prototyping.

UNIT – II
Tools for Design Thinking: Real-Time design interaction capture and analysis – Enabling efficient collaboration in digital space – Empathy for design – Collaboration in distributed Design
Practice using any AI tool:
Students are advised to use any AI tool like Otter AI etc for Transcribe interviews automatically and summarize user insights.

UNIT – III
Design Thinking in IT: Design Thinking to Business Process modelling – Agile in Virtual collaboration environment – Scenario based Prototyping
Practice using any AI tool:
Students are advised to use any AI tool like ChatGPT,Claude, Perplexity etc to define, synthesize, ideate and prototype.

UNIT – IV
DT For strategic innovations: Growth – Story telling representation – Strategic Foresight - Change – Sense Making - Maintenance Relevance – Value redefinition - Extreme Competition – experience design.
Practice using any AI tool:
Students are advised to use any AI tool like looka,logomkr,canva and figma etc for logo,prototyping and sketching.

UNIT – V

Design thinking workshop: Design Thinking Work shop Empathize, Design, Ideate, Prototype and Test.

Text Books:

1 Engineering Design, John.R.Karsnitz, Stephen O'Brien and John P. Hutchinson, Cengage learning (International edition) Second Edition, ISBN: 9781133708766
2 The Design of Business: Why Design Thinking is the Next Competitive Advantage, Roger Martin, Harvard Business Press, ISBN: 9781422177808

Reference Books:

1 Engineering Design Process, Yousef Haik and Tamer M.Shahin, Cengage Learning, Second Edition, ISBN: 978-8131529041
2 Book - Solving Problems with Design Thinking - Ten Stories of What Works (Columbia Business School Publishing) Hardcover – 20 Sep 2013 by Jeanne Liedtka (Author), Andrew King (Author), Kevin Bennett (Author), ISBN: 978-0231163569

Web Links:

1 www.tutor2u.net/business/presentations/. /Product lifecycle/default.html
2 https://docs.oracle.com/cd/E11108_02/otn/pdf/. /E11087_01.pdf

Universal Human Values

(Common to CE, EEE, ME, ECE, CSE, IT, AIML, CSE(DS), PT & Min.E)

	L	T	P	C
Course Code: 2501UC08	**1**	**1**	**0**	**2**

Pre Requisites: Universal Human Values (UHV-I) (desirable).

Course Objectives:

This course input is intended

1. To help the students appreciate the complementarity between 'VALUES' and 'SKILLS' and understand how ethical foundations enhance technical competence to achieve sustainable happiness, prosperity, and responsible professional growth.
2. To develop a Holistic perspective among students towards life and profession as well as towards happiness and prosperity based on a right understanding of the Human reality and the rest of Existence. Such a holistic perspective forms the basis of Universal Human Values (UHV) and movement towards value-based living in a natural way.
3. To highlight real-world implications of Holistic understanding of ethical human conduct, trustful and mutually fulfilling human behaviour and mutually enriching interaction with Nature.
4. To encourage the students to realise their full potential and act accordingly.
5. To assist the students to live with a feeling of relationship, harmony and co-existence.

Thus, this course is intended to provide a much-needed input in value education to the young enquiring minds.

UNIT I INTRODUCTION TO VALUE EDUCATION

Lecture 1: Holistic Development and the Role of Education (Right Understanding, Relationship and Physical Facility)
Lecture 2: Understanding Value Education
Tutorial 1: Practice Session PS1 - Sharing about Oneself
Lecture 3: Self-Exploration as the Process for Value Education
Lecture 4: Continuous Happiness and Prosperity – the Basic Human Aspirations
Tutorial 2: Practice Session PS2 - Exploring Human Consciousness
Lecture 5: Happiness and Prosperity – Current Scenario
Lecture 6: Method to Fulfill the Basic Human Aspirations
Tutorial 3: Practice Session PS3 - Exploring Natural Acceptance.

UNIT II HARMONY IN THE HUMAN BEING

Lecture 7: Understanding Human being as the Co-existence of the self and the body
Lecture 8: Distinguishing between the Needs of the self and the body
Tutorial 4: Practice Session PS4 - Exploring the difference of Needs of self and body
Lecture 9: The body as an Instrument of the self
Lecture 10: Understanding Harmony in the self
Tutorial 5: Practice Session PS5 - Exploring Sources of Imagination in the self
Lecture 11: Harmony of the self with the body
Lecture 12: Programme to ensure self-regulation and Health
Tutorial 6: Practice Session PS6 - Exploring Harmony of self with the body.

UNIT III HARMONY IN THE FAMILY AND SOCIETY

Lecture 13: Harmony in the Family – the Basic Unit of Human Interaction
Lecture 14: 'Trust' – the Foundational Value in Relationship
Tutorial 7: Practice Session PS7 - Exploring the Feeling of Trust
Lecture 15: 'Respect' – as the Right Evaluation
Tutorial 8: Practice Session PS8 - Exploring the Feeling of Respect
Lecture 16: Other Feelings, Justice in Human-to-Human Relationship
Lecture 17: Understanding Harmony in the Society
Lecture 18: Vision for the Universal Human Order
Tutorial 9: Practice Session PS9 - Exploring Systems to fulfil Human Goal

UNIT IV HARMONY IN THE NATURE/EXISTENCE

Lecture 19: Understanding Harmony in Nature
Lecture 20: Interconnectedness, self-regulation and Mutual Fulfilment among the Four Orders of Nature
Tutorial 10: Practice Session PS10 - Exploring the Four Orders of Nature
Lecture 21: Realizing Existence as Co-existence at All Levels
Lecture 22: The Holistic Perception of Harmony in Existence
Tutorial 11: Practice Session PS11 - Exploring Co-existence in Existence.

UNIT V IMPLICATIONS OF THE HOLISTIC UNDERSTANDING A LOOK AT PROFESSIONAL ETHICS

Lecture 23: Natural Acceptance of Human Values
Lecture 24: Definitiveness of (Ethical) Human Conduct
Tutorial 12: Practice Session PS12 - Exploring Ethical Human Conduct
Lecture 25: A Basis for Humanistic Education, Humanistic Constitution and Universal Human Order
Lecture 26: Competence in Professional Ethics
Tutorial 13: Practice Session PS13 - Exploring Humanistic Models in Education
Lecture 27: Holistic Technologies, Production Systems and Management Models-Typical Case Studies
Lecture 28: Strategies for Transition towards Value-based Life and Profession
Tutorial 14: Practice Session PS14 - Exploring Steps of Transition towards Universal Human Order

Course Outcomes:

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

CO1: Demonstrate an understanding of basic human aspiration and the concept of Natural Acceptance and apply them in personal and professional decision-making.
CO2: Develop self-awareness and practice self-regulation.
CO3: Recognize human-human relationship (Justice) and identify human goals in society.
CO4: Appreciate the harmony in nature and existence.
CO5: Develop as socially and ecologically responsible engineers in handling problems with sustainable solutions (user-friendly and eco-friendly).

Text Books:

1. R R Gaur, R Asthana, G P Bagaria, *A Foundation Course in Human Values and Professional Ethics*, 3rd Revised Edition, UHV Publications, Sarva Shubha Nyas, Kanpur, 2023. ISBN: 978- 81-957703-7-3 (Printed Copy) ISBN: 978-81-957703-6-6 (e-book)
2. R R Gaur, R Asthana, G P Bagaria, *Teachers' Manual for a Foundation Course in Human Values and Professional Ethics,* 3rd Revised Edition, UHV Publications, Sarva Shubha Nyas, Kanpur, 2023.

Reference Books:

1. *Jeevan Vidya: Ek Parichaya,* A Nagaraj, Jeevan Vidya Prakashan, Amarkantak, 1999
2. *Human Values*, A.N. Tripathi, New Age Intl. Publishers, New Delhi, 2004.
3. *The Story of Stuff* (Book), Annie Leonard, Free Press, 2010.
4. .*The Story of My Experiments with Truth*, Mohandas Karamchand Gandhi, Fingerprint! Publishing
5. *Small is Beautiful* - E. F Schumacher, Random House, 2011.
6. *Slow is Beautiful* - Cecile Andrews, New Society Publishers, 2006
7. *Economy of Permanence* - J C Kumarappa
8. *Vivekananda* - Romain Rolland

Online Resources

1. https://fdp-si.aicte-india.org/index.php

2. https://onlinecourses.swayam2.ac.in/aic22_ge23/preview
3. https://uhv.org.in/
4. https://www.youtube.com/@UniversalHumanValues/playlists

Skill Enhancement Courses (SEC)

Course Code	Course Name	Level	L	T	P	C	CIE	SEE	Total	Pre-requisite
2501CS17	Object Oriented Analysis & Design using UML	IC			2	2	50	50	100	ASE
2501AI34/ 2601CS01	Web Application Development using MERN Stack*/ Linux Programming#	AC			2	2	50	50	100	-
2501CS19	CI/CD using DevOps	AC			1	1	100	-	100	-
Total					**5**	**5**				

Object Oriented Analysis & Design using UML
(Common to CSE, IT AIML & CSE(DS))

	L	T	P	C
Course Code: 2501CS17	**0**	**0**	**2**	**2**

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

CO1: Explain the importance of system analysis and design in solving complex problems.
CO2: Compare object-oriented approach with traditional approach in system analysis and design.
CO3: Analyze the importance of modeling and design of various applications.
CO4: Compare various object relationships.
CO5: Show the role and function of each UML model in developing object- oriented software.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	2	1	-	-	3	-	-	2	2	-	-
CO2	3	1	-	-	2	-	-	2	2	-	-
CO3	3	-	2	-	2	-	-	2	2	-	-
CO4	3	2	1	-	2	-	-	2	2	-	-
CO5	2	3	-	-	2	-	-	2	2	-	-

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1	2	-
CO2	2	-
CO3	3	-
CO4	3	-
CO5	3	-

Practice:
List of Case Studies:

ATM Application.
Library Management System.
Online Book Shop.
Customer Support System.
Point of Sale.
Familiarization with Rational Rose

1. For each case study:
   1.1) Identify and analyze events
   1.2) Identify Use cases
2. For each case study:
   2.1) Develop event table

2.2) Identify &analyze domain classes
3. For each case study:
   3.1) Represent use cases and a domain class diagram using Rational Rose
   3.2) Develop CRUD matrix to represent relationships between use cases and problem domain classes
4. For each case study:
   4.1) Develop Use case diagrams
   4.2) Develop elaborate Use case descriptions &scenarios
5. For each case study
   5.1) develop system sequence diagrams
6. For each case study:
   6.1) Develop High-level sequence diagrams for each use case
   6.2) Develop Detailed Sequence Diagrams / Communication diagrams for each use case showing interactions among all the three-layer objects
7. For each case study:
   7.1) Develop Class Diagram (CD) from the Domain Model.
   7.2) Identify attributes, methods, visibility, associations, dependency, generalization, aggregation, composition, and inheritance relationships.
8. For each case study:
   8.1) Develop High-level Collaboration diagrams for each use case
   8.2) Develop detailed collaboration diagrams for each use case showing interactions among all the three-layer objects
9. For each case study
   9.1) Develop Use case Packages.
   9.2) Develop component diagrams.
   9.3) Develop sample diagrams for other UML diagrams state chart diagrams, activity diagrams and deployment diagrams.
10. Design UML models for the following application. Restaurant System.

Additional practice:
Design UML models for the following Case studies.

1. Banking Application System.
2. Railway Reservation System
3. Online Movie Ticket Booking System.

Corner Stone Project

Project 1. Smart Campus Management System

Implementation Mainly Focuses On

- Adding and managing student, faculty, and admin details.
- Managing multiple modules such as attendance, events, hostel, transport, and complaints.
- Recording and viewing student attendance information.
- Creating and managing campus events with notifications.
- Hostel room allocation and tracking student accommodation details.
- Managing transport routes, schedules, and student registrations.
- Complaint registration, tracking, and resolution process.

Documentation Must Contain

- Actors: Student, Admin, Faculty
- Use Cases: Mark Attendance, Register Complaint, Make Fee Payment
- Domain Classes: Student, Course, Bus, Complaint
- Sequence Diagrams: Attendance flow, Complaint resolution
- CRUD Matrix: Student, Course, Fee, Complaint

2. **Online Doctor Appointment & Health Record System**

Implementation mainly focuses on:

- Adding and managing patient, doctor, and admin details.
- Booking, rescheduling, and cancelling appointments.
- Maintaining digital health records and prescriptions.
- Viewing doctor availability and appointment schedules.
- Searching records using patient ID and doctor ID.
- Input validation (valid dates, unique IDs, required fields).
- Role-based login system (Patient, Doctor, Admin).

Documentation Must Contain

- Actors: Patient, Doctor, Admin
- Use Cases: Book Appointment, View Records, Issue Prescription
- Domain Classes: Patient, Doctor, Appointment, Prescription
- SSD: Appointment booking flow
- 3-layer Sequence: UI → Controller → Database

PROJECT 3: Smart Parking Management System

Implementation mainly focuses on:

- Managing parking slots and vehicle details.
- Checking real-time availability of parking slots.
- Booking and releasing parking slots.
- Generating parking tickets and billing details.
- Tracking slot status (Free, Reserved, Occupied).
- Searching vehicle details using vehicle number.
- Input validation (valid vehicle number, slot availability).
- Admin control for managing slots and pricing.
- Menu-driven interface for user interaction.

Documentation Must Contain:

- Actors: Driver, Admin
- Use Cases: Check Availability, Book Slot, Pay Fee
- Classes: Slot, Vehicle, Ticket, Payment
- State Chart: Slot (Free → Reserved → Occupied)
- Deployment Diagram: Mobile Application + Server + IoT Sensors

4. **Online Food Ordering & Delivery System**

Implementation mainly focuses on:

- Adding and managing customer, restaurant, and delivery details.
- Browsing menu and selecting food items.
- Adding items to cart and placing orders.
- Processing payments and generating order details.
- Tracking order status from preparation to delivery.
- Managing restaurant menu and order updates.
- Input validation (valid order details, required fields)

Documentation Must Contain:

- Actors: Customer, Restaurant, Delivery Agent
- Use Cases: Order Food, Track Order, Make Payment
- Classes: Order, Menu, Customer, Delivery
- Sequence: Order placement → Payment → Delivery

5. **E-Learning Platform (Course Management System)**

Implementation mainly focuses on:

- Adding and managing student, instructor, and admin details.
- Course creation, enrollment, and management.
- Uploading and accessing learning materials (videos, notes).
- Conducting quizzes and evaluating student performance.
- Tracking course progress and results.
- Searching courses and student records using IDs.
- Input validation (unique course ID, valid data entries).
- Role-based login system (Student, Instructor, Admin).

Documentation Must Contain:

- Actors: Student, Instructor, Admin

- Use Cases: Enrol Course, Upload Content, Take Quiz
- Classes: Course, Video, Quiz, Result
- Activity Diagram: Learning workflow

6. **Online Voting System**

Implementation mainly focuses on:

- Registering voters and managing candidate details.
- Conducting secure voting process.
- Preventing duplicate voting using unique voter ID.
- Recording and counting votes accurately.
- Displaying election results.
- Managing election phases (start, stop, result).
- Input validation (valid voter ID, one vote per user).
- Admin control for managing elections.
- Menu-driven interface for voting process.

Documentation Must Contain:

- Actors: Voter, Admin
- Use Cases: Register, Vote, Count Votes
- Classes: Voter, Candidate, Election
- Sequence Diagram: Voting process
- State Chart: Election states

7. **Railway Reservation System**

Implementation mainly focuses on:

- Adding and managing passenger and train details.
- Booking and cancelling train tickets.
- Seat allocation based on availability.
- Generating ticket details and reservation status.
- Searching tickets using passenger ID or ticket number.
- Managing waiting list and seat confirmation.
- Input validation (valid travel details, seat availability).
- Admin control for train schedules and seat management.

Documentation Must Contain:

- Actors: Passenger, Admin
- Use Cases: Book Ticket, Cancel Ticket
- Classes: Train, Ticket, Seat, Passenger
- CRUD Matrix: Ticket, Passenger, Payment

Text Books:

1 Object-oriented analysis and design using UML, Mahesh P. Matha, PHI, ISBN: 9788120333222
2 Head first object-oriented analysis and design, Brett D. McLaughlin, Gary Pollice, Dave West,O‟Reilly. ISBN:9780596008673

Reference Books:

1 Object-oriented analysis and design with the Unified process, John W. Satzinger, Robert B. Jackson, Stephen D. Burd, Cengage Learning. ISBN:978-0619216436
2 The Unified modeling language Reference manual, James Rumbaugh, Ivar Jacobson, Grady Booch, Addison Wesley. ISBN:9780321245625
3 Object Oriented Analysis & Design, Atul Kahate, The McGraw-Hills Companies. ISBN:9780070583764

Web Links:

1 https://onlinecourses.nptel.ac.in/noc24_cs40/preview
2 https://online.visual-paradigm.com/diagrams/solutions/free-uml-editor-online/
3 https://creately.com/lp/uml-diagram-tool/

Note:

- Students must submit the Certificate of Completion offered by any Industry.
- The corner stone project team size shall be four (4) students.
- Students may select any one of the above prescribed projects or a project of their own choice with the prior permission of the Course instructor.
- The SEE Lab shall be evaluated for 50 marks based on project implementation, oral presentation, 10-minute video presentation, and viva voce.
- The video presentation should consist of the working procedure of the project along with contribution of each student for a minimum of 2 minutes.

WEB Application Development using MERN Stack
(Common to CSE, IT, AIML & CSE (DS))
(Syllabus for the batch admitted in the year 2025)

	L	T	P	C
Course Code: 2501AI34	**0**	**0**	**2**	**2**

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

- **CO1:** Make use of HTML elements, CSS styles and their attributes for designing static web pages.
- **CO2:** Apply JavaScript to develop dynamic web pages and validate forms.
- **CO3:** Build a basic web server using Node.js and also working with Node Package Manager (NPM) & Make use of Typescript to optimize JavaScript code using strict type checking
- **CO4:** Make use of router, template engine and authentication using sessions to develop application in ExpressJS & components, props, stats and render data in ReactJS
- **CO5:** Build a single page application using RESTful APIs in ExpressJS. & Make use of MongoDB queries

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	2	2	3	-	2	-	-	1	2	-	2
CO2	2	3	2	-	2	-	-	1	2	-	2
CO3	2	2	3	1	2	-	-	1	2	-	2
CO4	2	3	2	2	2	-	-	1	2	-	2
CO5	2	2	3	1	2	-	-	1	2	-	2

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1	2	-
CO2	2	-
CO3	2	-
CO4	2	-
CO5	2	-

Practice:

1. Lists, Links, Images, Tables, Forms.
   a. Write a HTML program, to explain the working of lists.
   Note: It should have an ordered list, unordered list, nested lists and ordered list in an unordered list and definition lists.
   b. Write a HTML program, to explain the working of hyperlinks using <a> tag and href, target Attributes.
   Note: Use text to link https://www.aec.edu.in/
   Use image to link https://www.aec.edu.in/?p=Gallery

c. Create a HTML document that has your image and your friend's image with a specific height and width. Also when clicked on the images it should navigate to their respective profiles.
d. Write a HTML program, to explain the working of tables. (use tags: <table>, <tr>, <th>, <td> and attributes: border, rowspan, colspan)
e. Write a HTML program, to explain the working of forms by designing Registration form. (Note: Include text field, password field, number field, date of birth field.

2. HTML 5 and Cascading Style Sheets, Types of CSS & CSS Box Model
   a. Write a HTML program, that makes use of <article>, <aside>, <figure>, <figcaption>, <footer>, <header>, <main>, <nav>, <section>, , tags.
   b. Write a HTML program, to embed audio and video into HTML web page.
   c. Write a program to demonstrate the various ways you can reference a color in CSS.
   d. Write a program, to explain the importance of CSS Box model using
      i. Content ii. Border iii. Margin iv. padding

3. Selector forms & JavaScript-internal and & I/o
   a. Write a program to apply different types of selector forms
      i. Simple selector (element, id, class, group, universal)
      ii. Combinator selector (descendant, child, adjacent sibling, general sibling)
      iii. Write a program to embed internal and external JavaScript in a web page.
      iv. Write a program to explain the different ways for displaying output.

4. Javascript Pre-defined and User-defined Objects, Conditional & Loop Statements
   a. Write a program using document, window, math & string object properties and methods.
   b. Write a program which asks the user to enter three integers, obtains the numbers from the user and outputs HTML text that displays the larger number followed by the words "LARGER NUMBER" in an information message dialog. If the numbers are equal, output HTML text as "EQUAL NUMBERS".
   c. Write a program to print 1 to 10 numbers using for, while and do-while loops.

5. Javascript Functions and Events & Node.js
   a. Design a appropriate function should be called to display
      i. Factorial of that number
      ii. ii. Fibonacci series up to that number
      iii. Write a program to show the workflow of JavaScript code executable by creating web server in Node.js.
      iv. Write a program to transfer data over http protocol using http module.
      v. Create a text file src.txt and add the following content to it. (HTML, CSS, Javascript, Typescript, MongoDB, Express.js, React.js, Node.js)

6. Typescript
   a. Write a program to understand simple and special types.
   b. Write a program to understand function parameter and return types.

7. ExpressJS – Routing, HTTP Methods,Templating

a. Write a program to define a route, Handling Routes, Route Parameters, Query Parameters and URL building.
b. Write a program to accept data, retrieve data and delete a specified resource using http methods.
c. Write a program using templating engine.

8. ExpressJS – Cookies, Sessions, Authentication, Database.
   a. Write a program for session management using cookies and sessions.
   b. Write a program for user authentication.
   c. Write a program to connect MongoDB database using Mangoose and perform CRUD operations
9. ExpressJS – Templating, Form Data, Cookies, Sessions
   a. Write a program using templating engine.
   b. Write a program to work with form data.
   c. Write a program for session management using cookies and sessions.

10. ReactJS – Render HTML, JSX, Props and States
    a. Write a program to render HTML to a web page.
    b. Write a program for writing markup with JSX.
    c. Write a program to work with props and states.

11. ReactJS – Conditional Rendering, Rendering Lists, React Router
    a. Write a program for conditional rendering.
    b. Write a program for rendering lists.
    c. Write a program for routing to different pages using react router.

12. ReactJS – Hooks, Sharing data between Components, To-do list and Quiz & MongoDB – Installation, Configuration
    a. Write a program to understand the importance of using hooks.
    b. Write a program for sharing data between components.
    c. Design to-do list application.
    d. Install MongoDB and configure ATLAS

Cornerstone Project:

Project 1 : Student Management System (MERN CRUD App)

- **Implementation mainly focuses on**
  - Create, update, view, and delete student records using MongoDB, Express, React, Node.js.
- **Documentation must contain**
  - Problem Definition, Objectives, Analysis& Design, Implementation and Sample Code, Output screens, Conclusion & Enhancement

Project 2: Online Food Ordering System Using MERN

- **Implementation mainly focuses on**
  - Menu display, cart, orders, user login, admin panel.
- **Documentation must contain**

- Problem Definition, Objectives, Analysis& Design, Implementation and Sample Code, Output screens, Conclusion & Enhancement

Project 3: Personal Portfolio with Admin Dashboard (MERN + Authentication)

- **Implementation mainly focuses on**
  - Edit portfolio sections via a protected admin panel.
- **Documentation must contain**
  - Problem Definition, Objectives, Analysis& Design, Implementation and Sample Code, Output screens, Conclusion & Enhancement

Project 4: To-Do List Application with User Login (MERN + Sessions/JWT)

- **Implementation mainly focuses on**
  - Add, edit, delete tasks, user registration/login, MongoDB storage.
- **Documentation must contain**
  - Problem Definition, Objectives, Analysis& Design, Implementation and Sample Code, Output screens, Conclusion & Enhancement

Project 5: Event Registration Portal (HTML + CSS + JS + MERN Backend)

- **Implementation mainly focuses on**
  - Users register for events; admin can view/manage registrations.
- **Documentation must contain**
  - Problem Definition, Objectives, Analysis& Design, Implementation and Sample Code, Output screens, Conclusion &Enhancement

Project 6: E-Commerce Product Catalog Using MERN

- **Implementation mainly focuses on**
  - Product listings, categories, product details, cart basics.
- **Documentation must contain**
  - Problem Definition, Objectives, Analysis & Design, Implementation and Sample Code, Output screens, Conclusion & Enhancement

Project 7: College Information & Admission Portal (MERN Full Stack)

- **Implementation mainly focuses on**
  - Forms, tables, student admission data, admin management.
- **Documentation must contain**
  - Problem Definition, Objectives, Analysis & Design, Implementation and Sample Code, Output screens, Conclusion & Enhancement

Project 8: Blog Posting Platform Using MERN (CRUD + File Upload)

- **Implementation mainly focuses on**
  - Users create posts, add images, edit, comment, delete.
- **Documentation must contain**
  - Problem Definition, Objectives, Analysis & Design, Implementation and Sample Code, Output screens, Conclusion & Enhancement

Project 9: Movie Review and Rating System Using MERN

- **Implementation mainly focuses on**
  - Users can review movies, give ratings, view reviews list.

- **Documentation must contain**
  - Problem Definition, Objectives, Analysis & Design, Implementation and Sample Code, Output screens, Conclusion & Enhancement

Project 10: Employee Record Management (MERN CRUD)

- **Implementation mainly focuses on**
  - Employee details, salary, department, search filters.
- **Documentation must contain**
  - Problem Definition, Objectives, Analysis & Design, Implementation and Sample Code, Output screens, Conclusion & Enhancement

Project 11: Notes Application Using MERN with Authentication

- **Implementation mainly focuses on**
  - Create/edit notes, categorize them, private notes per user.
- **Documentation must contain**
  - Problem Definition, Objectives, Analysis & Design, Implementation and Sample Code, Output screens, Conclusion & Enhancement

Project 12. Online Quiz Application Using MERN (MCQ Based)

- **Implementation mainly focuses on**
  - Admin adds questions; users take quiz; store scores.
- **Documentation must contain**
  - Problem Definition, Objectives, Analysis & Design, Implementation and Sample Code, Output screens, Conclusion & Enhancement

Project 13: Library Book Management System

- **Implementation mainly focuses on**
  - Add books, issue/return system, student records, dashboards.
- **Documentation must contain**
  - Problem Definition, Objectives, Analysis & Design, Implementation and Sample Code, Output screens, Conclusion & Enhancement

Project 14: Hospital Appointment Booking System

- **Implementation mainly focuses on**
  - Users book appointments; doctors manage schedules.
- **Documentation must contain**

  Problem Definition, Objectives, Analysis & Design, Implementation and Sample Code, Output screens, Conclusion & Enhancement

Reference Books:

1 Programming the World Wide Web, Robet W Sebesta, Pearson, 7th Edition. ISBN 139789332518827

2 Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node, Vasan Subramanian, APress, O'Reilly, 2nd edition. ISBN 9781484226520

Web Links:

1 https://infyspringboard.onwingspan.com/en/app/toc/lex_17739732834840810000_shared/overview (HTML5)

2 https://infyspringboard.onwingspan.com/en/app/toc/lex_18109698366332810000_shared/overview (Javascript)

3 https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01384329019595161635073_shared/overview (Express Js)

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

Note:

- Students must submit the Certificate of Completion offered by any Industry.
- The capstone project team size shall be four (4) students.
- Students may select any one of the above prescribed projects or a project of their own choice with the prior permission of the Course instructor.
- The SEE – Lab shall be evaluated for 50 marks based on project implementation, oral presentation, 10-minute video presentation, report and viva voce.
- The video presentation should consists of the working procedure of the project along with contribution of each student for a minimum of 2 minutes.
- Finally, the Source code of the capstone Project has to be pushed into the Students GitHub repository.

Linux Programming
(Common to CSE, AIML,CSE(DS))
(Syllabus for the batch admitted in the year 2026)

	L	T	P	C
Course Code: 2601CS01	**0**	**0**	**2**	**2**

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

CO1: Demonstrate proficiency in Linux commands, file-system operations
CO2: Develop shell scripts using decision-making and looping constructs
CO3: Apply Linux utilities for file handling operations
CO4: Exhibit user management and file transfer operations
CO5: Develop shell scripts to access data using web urls.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	2	2	-	-	2	-	-	1	2	-	-
CO2	3	2	-	-	3	-	-	1	2	-	-
CO3	2	2	-	-	3	-	-	2	1	-	-
CO4	2	2	-	-	1	-	-	1	1	-	-
CO5	2	2	-	-	2	-	-	1	1	-	-

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1	1	2
CO2	2	1
CO3	2	1
CO4	1	2
CO5	2	1

Practice:

Week 1: Practice the following basic Linux commands
Basic commands: pwd, ls, mkdir, rmdir, chmod, chown, time, cp, rm, man, echo, touch, cat, vi, cal, df, du, nano, vim, fallocate, read, bash.
Week 2: Observe the usage of the following Networking commands: useradd, userdel, passwd, df, du, ps, uname, hostname, systemctl, find, mount watch, jobs, kill, shutdown, ping, wget, curl, scp, rsync, ip, netstat, traceroot, nslookup, dig, history, ln, alias, unalias
Week 3: Excersie how to execute sudo command for adding and deleting users
Root priviliges: sudo, sudo adduser, sudo deluser, sudo deluser, sudo mv, sudo gedit, uname, sudo apt-get update, sudo apt-get install, sudo apt-get upgrade, sudo apt-get remove, sudo apt-get purge, sudo apt-get autoremove, sudo apt-get clean, sudo apt-get autoclean, sudo apt-get check
Week 4: if-fi loop variants

1. Write a shell script to display your name by taking input from the user.
2. Write a shell script to calculate SGPA using conditional statements.
3. Write a shell script to find the factorial of a number by taking input from the user.

Week 5: For and case-esac

1. Write a shell script using for loop to print numbers of the user's choice by accepting input at the command prompt.
2. Write a shell script to create menu using case and esac.

Week 6: File operations

1. Write a shell script to extract first column of data from .csv file.
2. Write a shell script to copy file to another using 'cp' command.

Week 7: Editing file

1. Write a shell script to create and edit a file using vi and nano.
2. Write a shell script to a) find b) replace c) first and third columns of a .csv file using sed, find and cut commands.
3. Write a shell script to remove duplicate lines in a file.

Week 8: Pattern matching using filters

1. Write a shell script to search for a pattern using 'grep' with all possible flags and wildcards.
2. Write a shell script to sort lines of text which is stored in a .csv file.
3. Write a shell script using while loop to display week days.

Week 9: Changing the location of file from one location to another

1. Write a shell script to link and unlink a text file using ln and unlink commands.
2. Write a shell script to check username and password of the user logged in.

Week 10: User previliges using sudo command

1. Illustrate 'sudo' to show administrative privileges with all possible options.
2. Illustrate 'su' command to show user logged in with possible options.

Week 11: Accessing file from web and perform operations

1. Write a shell script to display contents of a file from web using curl.
2. Write a shell script to open a 'google.com' from the shell prompt using 'xdg-open'.

Week 12: Installing software using sudo apt-get install

1. Install python in linux terminal and execute simple python program.
2. Install 'C' on linux platform and execute simple program.

Week 13: Transfer of files from local to remote

1. Install VMWARE and run 'C' program
2. Install VMWARE and run simple Java program.

Additional Practice

1. Write a shell script to create menus for illustrating inter process communication for displaying student details like marks, attendance, list of students in each branch etc.
2. Write a c program using signals in UNIX for checking authenticated user or not by checking with the password file.

Text Books:

1. Red Hat Enterprise Linux 9 , Vishesh Kumar, Gitanjali Mehra, BPB Publications.
2. Linux Essentials – Christine Bresnahan and Richard Blum, Wiley Publications.

Reference Books:

1. The Linux Command Line: A Complete Introduction (2nd Edition) – William E. Shotts Jr. Linux Command Line and Shell Scripting Bible (4th Edition) – Richard Blum and Christine Bresnahan.
2. Linux Administration: A Beginner's Guide (8th Edition) – Wale Soyinka.
3. UNIX and Linux System Administration Handbook (5th Edition) – Evi Nemeth, Garth Snyder, Trent R. Hein, Ben Whaley, Dan Mackin.
4. How Linux Works: What Every Superuser Should Know (3rd Edition) – Brian Ward.
5. Red Hat System Administration I (RH124) – Red Hat Press.

Web Links:

1. https://docs.redhat.com/en/documentation/red_hat_enterprise_linux/9
2. https://www.gnu.org/software/bash/manual/
3. https://labex.io/linuxjourney
4. https://www.freecodecamp.org/news/tag/linux/
5. https://linuxize.com/

CI/CD using Devops
(Common to CSE, IT, AIML & CSE (DS))

Semester:	**L**	**T**	**P**	**C**
Course Code: 2501CS19	**0**	**0**	**1**	**1**

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

CO1: Summarize the life cycle & principles of agile software development
CO2: Develop the fundamentals programs using Devops
CO3: Analyze the adoption of Devops in Projects and Process
CO4: Demonstrate the types of continuous integration and continuous delivery in Devops
CO5: Build an automated CICD pipeline using a stack of tools
CO6: Develop basic branching and merging in GIT

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	1	3	1	-	2	-	-	2	2	-	1
CO2	1	2	1	-	3	-	-	2	2	-	1
CO3	1	3	2	2	2	-	-	2	2	-	1
CO4	-	1	3	-	2	-	-	2	2	-	1
CO5	2	1	3	-	3	-	-	2	2	-	2
CO6	2	3	2	-	3	-	-	2	2	-	1

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1	1	-
CO2	1	-
CO3	2	-
CO4	2	-
CO5	3	-
CO6	2	-

Practice:

1. Get an understanding of the stages in software development lifecycle, the process models, values and principles of agility and the need for agile software development. This will enable you to work in projects following an agile approach to software development.

2. Get a working knowledge of using extreme automation through XP programming practices of test first development, refactoring and automating test case writing.

3. It is important to comprehend the need to automate the software development lifecycle stages through DevOps. Gain an understanding of the capabilities required to implement DevOps, continuous integration and continuous delivery practices.

4. Configure the web application and Version control using Git using Git commands and version control operations.

5. Configure a static code analyzer which will perform static analysis of the web application code and identify the coding practices that are not appropriate. Configure the profiles and dashboard of the static code analysis tool.

6. Write a build script to build the application using a build automation tool like Maven. Create a folder structure that will run the build script and invoke the various software development build stages. This script should invoke the static analysis tool and unit test cases and deploy the application to a web application server like Tomcat.

7. Configure the Jenkins tool with the required paths, path variables, users and pipeline views.

8. Configure the Jenkins pipeline to call the build script jobs and configure to run it whenever there is a change made to an application in the version control system. Make a change to the background color of the landing page of the web application and check if the configured pipeline runs.

9. Create a pipeline view of the Jenkins pipeline used in Exercise 8. Configure it with user defined messages.

10. In the configured Jenkins pipeline created in Exercise 8 and 9, implement quality gates for static analysis of code.

11. In the configured Jenkins pipeline created in Exercise 8 and 9, implement quality gates for static unit testing.

12. In the configured Jenkins pipeline created in Exercise 8 and 9, implement quality gates for code coverage.

Additional Practice:

1. To fetch and synchronize git repository
2. To perform basic branching and merging in Git
3. To implement form validation

Reference Books:

1 Learning Continuous Integration with Jenkins: A beginner's guide to implementing Continuous Integration and Continuous Delivery using Jenkins - Nikhil Pathania ,Packt publication [https://www.amazon.in/Learning-Continuous-Integration-Jenkins-Pat, ISBN: 978-1785284830

2 Jenkins 2 – Up and Running: Evolve Your Deployment Pipeline for Next Generation Automation – Brent Laster, O'Reilly publication, ISBN: 978-1491979594 [https://www.amazon.in/Jenkins-2-Running-Brent-Laster/dp/1491979593]

Web Links:

1 https://www.geeksforgeeks.org/what-is-ci-cd/
2 https://github.com/nkatre/Free-DevOps-Books-1/blob

Value Added Courses (VAC)

Course Code	Course Name	Level	L	T	P	C	CIE	SEE	Total	Pre-requisite
2501CS04	Internet of Things	**FC**			1	1	50	50	100	-
2501AI06	Data Analysis Essentials	IC			2	2	50	50	100	PP
2501UC11	Employability Skills-I	FC			3	0	100	-	100	-
2501UC13	Employability Skills-II	FC			3	0	100	-	100	ES-I
2501UC14	Employability Skills-III	IC			3	0	100	-	100	ES-II
2501UC15	Employability Skills-IV	IC			3	0	100	-	100	ES-III
2501UC16	Employability Skills-V	AC			3	1	100	-	100	ES-IV
Total					**18**	**4**				

Internet of Things

(Common to EEE, ME, ECE, CSE, IT, AIML, CSE(DS), PT & Min.E)

	L	T	P	C
Course Code: 2501CS04	**0**	**0**	**1**	**1**

Course Outcomes:

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

CO1: Identify and select appropriate sensors, actuators, and boards (Arduino/Pi) for specific tasks.
CO2: Execute digital and analog interfacing of peripherals with embedded platforms.
CO3: Develop control algorithms and decision-making logic for real-time automation.
CO4: Configure cloud platforms and APIs for data transmission and visualization.
CO5: Design and build functional end-to-end IoT prototypes for real-world applications.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	2	1			1						
CO2	2	2	1		2						
CO3	1	2	2	1							
CO4	1	1		1	2						1
CO5	1	2	2	2	2	1	1	1	2	2	2

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1		
CO2		
CO3		
CO4		
CO5		

Practice:

1. Study of Arduino UNO and Raspberry Pi boards including architecture, pin configuration, power supply, communication interfaces and development environments.
2. Study of commonly used sensors and actuators such as temperature sensor, moisture sensor, LDR, ultrasonic sensor, IR sensor, PIR sensor, LED and motor. Identify their working principles and applications.
3. Design and implement basic digital input–output operations using Arduino by controlling LED/buzzer.
4. Interface an analog sensor (LDR) with Arduino and display sensor values on Serial Monitor.
5. Interface a digital sensor (Ultrasonic sensor) with Arduino and display measured values on Serial Monitor.
6. Develop conditional logic using Arduino to control LED or buzzer based on sensor threshold values.

7. Interface a servo motor with Arduino and control its angular position based on sensor input to demonstrate automatic gate or smart irrigation model.
8. Design and implement a multi-sensor monitoring system using Arduino to read analog and digital sensors simultaneously and display results on Serial Monitor.
9. Create a cloud platform account, explore available IoT services and register a device (Thing) on the platform.
10. Demonstration of IoT based system to upload sensor data to cloud platform and visualize the received data in graphical form.

Additional Practice (Any 2 from the following)

1. **Smart Lighting System**: Design a lighting system that automatically turns ON/OFF or dims lights based on human movement or surrounding light conditions to demonstrate energy saving.
2. **Intelligent Traffic Control System**: Develop an automated traffic signal system for busy roads with priority control for emergency vehicles (such as ambulances).
3. **Smart Parking System**: Create a sensor-based parking system to detect available slots and provide parking guidance or remote monitoring.
4. **Air Pollution Monitoring System**: Build an IoT-based setup using gas or environmental sensors to measure air quality and indicate whether pollution levels are safe or dangerous.
5. **Bluetooth-Based Actuator Control**: Control motors or other actuators connected to a development board using Bluetooth communication.

Reference Books:

1 An Introduction to Internet of Things, Connecting devices, Edge Gateway and Cloud with Applications, Rahul Dubey, Cengage, 2019. ISBN: 9789353501020
2 Adrian McEwen, Designing the Internet of Things, Wiley Publishers. ISBN:978-1118430620
3 IoT Fundamentals, Networking Technologies, Protocols and Use Cases for the Internet of Things, David Hanes, Gonzalo Salgueiro, Patrick Grossetette, rob Barton, Jerome Henry, CISCO, Pearson, 2018. ISBN:9789386873743
4 Internet of Things, by Vengalapudi Appala Konda and Prof. Sangareddy B. Krutakoti, Scientific International Publishing House, First Edition, 2024. ISBN: 978-93-6132-958-6.

Web Links:

1 https://iotvirtuallab.github.io/vlab/Experiments/index.html
2 https://onlinecourses.nptel.ac.in/noc21_cs17/preview
3 https://www.electronicsforu.com/electronics-projects/internet-of-things-iot
4 https://www.coursera.org/specializations/iot
5 https://projecthub.arduino.cc/
6 https://www.edureka.co/iot-certification-training

Data Analysis Essentials
(Common to CSE, IT, AIML & CSE (DS))

	L	T	P	C
Course Code: 2501AI06	**0**	**0**	**2**	**2**

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

CO1: Perform various operations on NumPy arrays.
CO2: Importing data from different file formats using pandas
CO3: Draw different types of charts using matplotlib.
CO4: Demonstrate web scraping using python.
CO5: Design preprocessing techniques on loan prediction dataset.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	2	2	-	-	2	-	-	1	2	-	-
CO2	3	2	-	-	3	-	-	1	2	-	-
CO3	2	2	-	-	3	-	-	2	1	-	-
CO4	2	2	-	-	1	-	-	1	1	-	-
CO5	2	2	-	-	2	-	-	1	1	-	-

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1	1	2
CO2	2	-
CO3	2	-
CO4	1	2
CO5	2	-

Practice:
Week 1: Creating a NumPy Array

a. Basic ndarray
b. Array of zeros
c. Array of ones
d. Random numbers in ndarray
e. An array of your choice
f. Imatrix in NumPy
g. Evenly spaced ndarray

Week 2: The Shape and Reshaping of NumPy Array

a. Dimensions of NumPy array
b. Shape of NumPy array
c. Size of NumPy array
d. Reshaping a NumPy array
e. Flattening a NumPy array

f. Transpose of a NumPy array

Week 3: Expanding and Squeezing a NumPy Array

a. Expanding a NumPy array
b. Squeezing a NumPy array
c. Sorting in NumPy Arrays

Week 4: Indexing and Slicing of NumPy Array

a. Slicing 1-D NumPy arrays
b. Slicing 2-D NumPy arrays
c. Slicing 3-D NumPy arrays
d. Negative slicing of NumPy arrays

Week 5: Stacking and Concatenating Numpy Arrays

a. Stacking ndarrays
b. Concatenating ndarrays
c. Broadcasting in Numpy Arrays

Week 6: Perform following operations using pandas

a. Creating dataframe
b. concat()
c. Setting conditions
d. Adding a new column

Week 7: Perform following operations using pandas

a. Filling NaN with string
b. Sorting based on column values
c. groupby()

Week 8: Read the following file formats using pandas

a. Text files
b. CSV files
c. Excel files
d. JSON files

Week 9: Read the following file formats

a. Pickle files
b. Image files using PIL
c. Multiple files using Glob
d. Importing data from database

Week 10: Demonstrate web scraping using python

Week 11: Perform following preprocessing techniques on loan prediction dataset

a. Feature Scaling
b. Feature Standardization
c. Label Encoding
d. One Hot Encoding

Week 12: Perform following visualizations using matplotlib

a. Bar Graph
b. Pie Chart
c. Box Plot
d. Histogram
e. Line Chart and Subplots
f. Scatter Plot.

Additional Practice:

1. Write a program that reads a list of temperatures from a file called temps.txt, converts those temperatures to Fahrenheit, and writes the results to a file called ftemps.txt
2. Write a program to find the greatest number that can be formed by using given set of numbers.
3. Write a program to find sum of digits of a number till you get single digit sum.
   Example:
   Input: 142 (Hint: 1+4+2=7)
   Output: 7
   Input: 4683 (Hint: 4+6+8+3=21 =>2+1=3)
   Output: 3
4. Write a program to count how many times each word present in a file.

Cornerstone Project:

PROJECT 1: Loan Prediction Data Analysis System
Project Implementation, Documentation and Viva Voce

- **Implementation mainly focuses on**
  - Loading and exploring loan dataset using pandas
  - Handling missing values (NaN filling, cleaning)
  - Applying preprocessing techniques:
  - Feature Scaling
  - Standardization
  - Label Encoding & One Hot Encoding
  - Performing exploratory data analysis (EDA)
  - Visualizing insights using matplotlib (bar charts, histograms, etc.)
  - Basic prediction logic using processed data
- **Documentation must contain**
  - Objective & problem definition
  - Dataset description (features & target variable)
  - Data preprocessing steps explanation
  - Visualization results and insights
  - Workflow diagram (Data → Preprocess → Analyze → Visualize)
  - Screenshots of outputs

- Enhancements (feature selection, correlation analysis)

PROJECT 2: COVID-19 Data Analysis Dashboard

- **Implementation mainly focuses on**
  - Importing CSV/JSON datasets using pandas
  - Cleaning and organizing real-world data
  - Grouping data (country/state-wise analysis using groupby())
  - Time-series analysis of cases
  - Visualization:
  - Line charts (trend analysis)
  - Bar graphs (comparison)
  - Pie charts (distribution)
  - Generating summary statistics
- **Documentation must contain**
  - Objective
  - Dataset source and structure
  - Data cleaning steps
  - Graphical analysis and interpretation
  - Module explanation (Load → Clean → Analyze → Visualize)
  - Screenshots
  - Enhancements (live data scraping)

PROJECT 3: Student Performance Data Analysis System

- **Implementation mainly focuses on**
  - Creating and managing student dataset using pandas
  - Performing operations:
  - Sorting
  - Filtering
  - Adding calculated columns (average, grade)
  - Identifying performance trends
  - Visualization using matplotlib:
  - Bar graph (marks comparison)
  - Pie chart (grade distribution)
  - Generating reports
- **Documentation must contain**
  - Objective
  - Dataset schema
  - Data processing logic
  - Visualization explanation
  - Flowchart of system
  - Screenshots
  - Enhancements (subject-wise analysis, topper prediction)

PROJECT 4: Web Scraping and Product Price Analysis System

- **Implementation mainly focuses on**
  - Scraping product data (price, name, rating) using Python
  - Storing scraped data into pandas Data Frame

- Cleaning and structuring scraped data
- Performing analysis (price comparison, trends)
- Visualization:
  - Bar charts (price comparison)
  - Scatter plots (rating vs price)

- **Documentation must contain**
  - Objective
  - Website/data source details
  - Scraping methodology
  - Data cleaning process
  - Analysis and results
  - Screenshots
  - Enhancements (multiple site comparison)

PROJECT 5: Sales Data Analysis System

- **Implementation mainly focuses on**
  - Reading data from CSV/Excel files
  - Data preprocessing and cleaning
  - Performing operations:
  - Groupby (region/product sales)
  - Aggregation (total, average sales)
  - Visualization:
  - Line chart (sales trends)
  - Bar graph (region-wise sales)
  - Histogram (sales distribution)
- **Documentation must contain**
  - Objective
  - Dataset structure
  - Data processing steps
  - Graph interpretation
  - Module flow (Input → Process → Output)
  - Screenshots
  - Enhancements (profit analysis, forecasting basics)

PROJECT 6: File Data Analyzer System

- **Implementation mainly focuses on**
  - Reading multiple file formats:
  - Text, CSV, JSON, Excel
  - Performing operations:
  - Word count
  - Data filtering
  - Statistical analysis
  - Handling multiple files using glob
  - Visualization of extracted data
- **Documentation must contain**

- Objective
- File formats handled
- Processing logic
- Output results
- Screenshots
- Enhancements (automation, batch processing)

PROJECT 7: Social Media Data Analysis System

- **Implementation mainly focuses on**
  - Web scraping social media/text data
  - Cleaning text data
  - Word frequency analysis
  - Visualization:
  - Bar chart (top words)
  - Pie chart (category distribution)
  - Basic sentiment-style categorization
- **Documentation must contain**
  - Objective
  - Data source
  - Preprocessing steps
  - Analysis results
  - Visualization explanation
  - Screenshots
  - Enhancements (real-time scraping, dashboards)

PROJECT 8: Weather Data Analysis System

- **Implementation mainly focuses on**
  - Reading weather dataset (CSV/API/web scraping)
  - Cleaning and preprocessing data
  - Analyzing temperature trends
  - Converting units (Celsius ↔ Fahrenheit)
  - Visualization:
  - Line chart (temperature trends)
  - Histogram (distribution)
- **Documentation must contain**
  - Objective
  - Dataset description
  - Data transformation steps
  - Analysis results
  - Graph interpretation
  - Screenshots
  - Enhancements (forecasting basics)

Text Books:

1	Python Data Science Handbook, Jake VanderPlas, 2nd Edition,2022, ISBN**:** 978-1098121228
2	Data Analysis from Scratch with Python: Step By Step Guide, Peters Morgan,2018, ISBN**:** 978-1725678095

Reference Books:

1	Python for Data Analysis, Wes McKinney, 3rd Edition 2022, ISBN**:** 978-1098104030
2	An Introduction to Statistical Learning: With Applications in Python, Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani, Jonathan Taylor, 1st Edition ,2023, ISBN**:** 978-3031387463

Web Links:

1	https://www.analyticsvidhya.com/blog/2020/04/the-ultimate-numpy-tutorial-for-data-science-beginners/
2	https://www.analyticsvidhya.com/blog/2021/07/data-science-with-pandas-2-minutes-guide-to-key-concepts/

Note:

- Students must submit the Certificate of Completion offered by any Industry.
- The capstone project team size shall be four (4) students.
- Students may select any one of the above prescribed projects or a project of their own choice with the prior permission of the Course instructor.
- The SEE – Lab shall be evaluated for 50 marks based on project implementation, oral presentation, 10-minute video presentation, and viva voce.
- The video presentation should consist of the working procedure of the project along with contribution of each student for a minimum of 2 minutes.

Employability Skills- I

(Common to CE, EEE, ME, ECE, CSE, IT, AIML, CSE(DS), PT, Min.E & Agri.E)

	L	T	P	C
Course Code: 2501UC11	**0**	**0**	**3**	**0**

Course Outcomes:

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

CO1:	Apply fundamental arithmetic concepts such as number systems, LCM & HCF, ratio & proportion, and averages to solve quantitative problems efficiently.
CO2:	Analyze number and letter series to identify logical patterns and predict subsequent elements.
CO3:	Select the option by evaluating relationships in analogy-based problems and detecting inconsistencies in logical sequences
CO4:	Demonstrate effective verbal communication skills by using appropriate parts of speech and tone in various interpersonal and professional contexts.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	2	3	-	-	1	-	-	-	-	-	-
CO2	1	3	-	2	2	-	-	-	-	-	-
CO3	-	3	-	2	1	-	-	-	-	-	-
CO4	-	-	-	-	-	-	-	2	3	-	1

Aptitude:

Number System, LCM & HCF, Ratio and Proportion, Averages

Reasoning:

Number Series, Letter Series, Number Analogy, Letter Analogy, Odd Man Out, Logical Sequence of Words.

Verbal:

Introduction to soft skills, how to improve communication? Parts of Speech, Mind your language towards better English, Vocabulary Expansion

Text Books

1. Quantitative Aptitude –Dr. R. S. Aggarwal, S CHAND, **ISBN:** 9789355012326
2. A Modern Approach to Verbal and Non-Verbal Reasoning – Dr. R. S. Aggarwal, **ISBN**-13: 978-9352832163.
3. Quick Learning Objective General English – Dr. R. S. Aggarwal, S CHAND, **ISBN**-13: 978-8121922111.

Reference Books:

1. Quantitative Aptitude – Abhijit Guha Mc Graw Hill Publications, **ISBN-13** 9789389957426.

2 How To Crack Test of Reasoning Verbal , Non-Verbal & Analytical – Jaikishan and Premkishan,Arihant Publications, NEW EDITION 2024, **ASIN : B0CRQ9BVBC**.
3 A New Approach to Objective English – R. S. Dhillon DGP Publications, ISBN-13: **978-8186651032**.

Web Links:

1 www.indiabix.com
2 www.bankersadda.com

Employability Skills- II

(Common to CE, EEE, ME, ECE, CSE, IT, AIML, CSE(DS), PT, Min.E & Agri.E)

	L	T	P	C
Course Code: 2501UC13	**0**	**0**	**3**	**0**

Course Outcomes:

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

CO1: Apply percentage, profit & loss, and partnership concepts to solve real-world quantitative problems.

CO2: Analyze age-related problems to develop logical problem-solving strategies.

CO3: Evaluate logical reasoning questions including ranking, directions, and alphabet tests for accurate decision-making.

CO4: Solve pattern-based problems in coding and decoding to enhance analytical skills.

CO5: Demonstrate effective written communication using correct grammar and structured presentation techniques

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	3	3	-	-	2	-	-	-	-	-	-
CO2	2	3	2	-	1	-	-	-	-	-	-
CO3	-	3	-	2	2	-	-	-	-	-	-
CO4	-	3	-	2	2	-	-	-	-	-	-
CO5	-	-	-	-	-	-	-	2	2	-	2

Aptitude:

Problems on Ages, Partnership, Percentages, Profit and Loss

Reasoning:

Coding and Decoding, Ranking Test, Alphabet Test, Direction Test

Verbal:

Written communication skill practice, Grammatical use, Concept of 4 step method for presentation, Present Tense

Text Books

1. Quantitative Aptitude –Dr. R. S. Aggarwal, S CHAND, **ISBN** · 9789355012326
2. A Modern Approach to Verbal and Non-Verbal Reasoning – Dr. R. S. Aggarwal, **ISBN**-13. 978-9352832163.
3. Quick Learning Objective General English – Dr. R. S. Aggarwal, S CHAND, **ISBN**-13. 978-8121922111.

Reference Books:

1. Quantitative Aptitude – Abhijit Guha Mc Graw Hill Publications, **ISBN-13** 9789389957426.
2. How To Crack Test of Reasoning Verbal , Non-Verbal & Analytical – Jaikishan and

Premkishan,Arihant Publications, NEW EDITION 2024, **ASIN : B0CRQ9BVBC**
3 A New Approach to Objective English – R. S. Dhillon DGP Publications, isbn-13:978-8186651032

Web Links:

1 www.indiabix.com
2 www.bankersadda.com

Employability Skills- III

(Common to CE, EEE, ME, ECE, CSE, IT, AIML, CSE(DS), PT, Min.E & Agri.E)

	L	T	P	C
Course Code: 2501UC14	**0**	**0**	**3**	**0**

Course Outcomes:

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

CO1: Solve financial and quantitative aptitude problems using concepts of simple and compound interest.

CO2: Apply logical time-work frameworks to solve real-time problems related to work efficiency and pipe systems .

CO3: Analyze and interpret problems related to blood relations, clocks, calendars, and coded inequalities.

CO4: Evaluate spatial and logical reasoning through cubes, dice, and symbolic interpretation problems.

CO5: Demonstrate effective communication using correct grammar, and participate actively in discussions and comprehension activities.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	2	3	-	-	1	-	-	-	-	-	-
CO2	3	3	-	-	1	-	-	-	-	-	-
CO3	-	3	-	2	1	-	-	-	-	-	-
CO4	-	3	-	2	2	-	-	-	-	-	-
CO5	-	-	-	-	-	-	-	2	3	-	2

Aptitude:

Simple Interest, Compound Interest, Time and Work, Pipes and Cisterns

Reasoning:

Blood Relations, Calendar, Clocks, Cubes and Dice, Coded Inequalities

Verbal:

Grammar in use, Group discussion, Reading Comprehension, Past Tense, Future Tense

Text Books

1 Quantitative Aptitude –Dr. R. S. Aggarwal, S CHAND, ISBN: 9789355012326

2 A Modern Approach to Verbal and Non-Verbal Reasoning – Dr. R. S. Aggarwal, ISBN-13: 978-9352832163

3 Quick Learning Objective General English – Dr. R. S. Aggarwal, S CHAND, ISBN-13: 978-8121922111

Reference Books:

1 Quantitative Aptitude – Abhijit Guha Mc Graw Hill Publications, ISBN-13: 9789389957426

2 How To Crack Test of Reasoning Verbal , Non-Verbal & Analytical – Jaikishan and Premkishan,Arihant Publications, NEW EDITION 2024, **ASIN : B0CRQ9BVBC**.
3 A New Approach to Objective English – R. S. Dhillon DGP Publications, ISBN-13: 978-8186651032

Web Links:

1 www.indiabix.com
2 www.bankersadda.com

Employability Skills- IV

(Common to CE, EEE, ME, ECE, CSE, IT, AIML, CSE(DS), PT, Min.E & Agri.E)

	L	T	P	C
Course Code: 2501UC15	**0**	**0**	**3**	**0**

Course Outcomes:

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

CO1: Apply concepts of time, speed, distance, trains, boats and streams to solve motion-related aptitude problems.

CO2: Solve mensuration problems involving surface area and volume in real-world contexts.

CO3: Analyze logical reasoning questions involving Venn diagrams and syllogisms to make valid conclusions.

CO4: Interpret seating arrangements and solve non-verbal reasoning questions involving patterns and spatial reasoning.

CO5: Demonstrate effective communication through structured self-introduction, emails, letters, reports, and grammar usage.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	3	3	-	-	2	-	-	-	-	-	-
CO2	3	3	-	-	2	-	-	-	-	-	-
CO3	-	3	-	2	2	-	-	-	-	-	-
CO4	-	3	-	2	2	-	-	-	-	-	-
CO5	-	-	-	-	-	-	-	2	2	-	2

Aptitude:

Time, Speed and Distance, Problems on Trains, Boats and Streams, Mensuration - I, Mensuration – II

Reasoning:

Venn Diagrams, Syllogisms, Non - Verbal Reasoning, Seating Arrangement

Verbal:

Grammatical use, Self-introduction, Letters, E-Mail & Report writing, Error correction, Effective Communication

Text Books

1. Quantitative Aptitude –Dr. R. S. Aggarwal, S CHAND, ISBN: 9789355012326
2. A Modern Approach to Verbal and Non-Verbal Reasoning – Dr. R. S. Aggarwal, ISBN-13: 978-9352832163
3. Quick Learning Objective General English – Dr. R. S. Aggarwal, S CHAND,ISBN-13: 978-8121922111

Reference Books:

1 Quantitative Aptitude – Abhijit Guha Mc Graw Hill Publications, ISBN-13: 9789389957426
2 How To Crack Test of Reasoning Verbal , Non-Verbal & Analytical – Jaikishan and Premkishan,Arihant Publications, NEW EDITION 2024, **ASIN : B0CRQ9BVBC**
3 A New Approach to Objective English – R. S. Dhillon DGP Publications, ISBN-13: **978-8186651032**.

Web Links:

1 www.indiabix.com
2 www.bankersadda.com

Employability Skills- V

(Common to CE, EEE, ME, ECE, CSE, IT, AIML, CSE(DS), PT & Min.E)

	L	T	P	C
Course Code: 2501UC16	**0**	**0**	**3**	**1**

Course Outcomes:

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

CO1: Apply principles of permutations, combinations, probability, and logarithms to solve quantitative aptitude problems.

CO2: Interpret and analyze data from charts, graphs, and statistical tables for decision-making and accuracy.

CO3: Evaluate logical reasoning statements involving arguments, assumptions, conclusions, and data sufficiency.

CO4: Solve puzzle and eligibility test problems using analytical and critical thinking skills.

CO5: Demonstrate effective interpersonal, communication, and time management skills for successful personal and professional interactions.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	3	3	-	-	2	-	-	-	-	-	-
CO2	3	3	-	-	2	-	-	-	-	-	-
CO3	-	3	-	2	2	-	-	-	-	-	-
CO4	-	3	-	2	2	-	-	-	-	-	-
CO5	-	-	-	-	-	-	-	2	3	-	2

Aptitude:

Permutations and Combinations, Probability, Data Interpretation, Logarithms, Statistics

Reasoning:

Puzzle Tests, Eligibility Test, Data Sufficiency, Statements – Arguments, Statements – Assumptions, Statements - Course of Action, Statements – Conclusions

Verbal:

Interview skills, Grammar in use, Interpersonal Skills, Negotiation Skills, Social Skills, Problem-Solving Skills, Time Management Skills

Text Books

1. Quantitative Aptitude –Dr. R. S. Aggarwal, S CHAND, **ISBN** · 9789355012326.
2. A Modern Approach to Verbal and Non-Verbal Reasoning – Dr. R. S. Aggarwal, ISBN-13: 978-9352832163
3. Quick Learning Objective General English – Dr. R. S. Aggarwal, S CHAND, ISBN-13: 978-8121922111

Reference Books:

1. Quantitative Aptitude – Abhijit Guha Mc Graw Hill Publications, **ISBN-13** 9789389957426.
2. How To Crack Test of Reasoning Verbal , Non-Verbal & Analytical – Jaikishan and

Premkishan, Arihant Publications, NEW EDITION 2024, **ASIN : B0CRQ9BVBC**.
3 A New Approach to Objective English – R. S. Dhillon DGP Publications, ISBN-13: **978-8186651032**.

Web Links:

1 www.indiabix.com
2 www.bankersadda.com

Mandatory Courses (MC)

Course Code	Course Name	Level	L	T	P	C	CIE	SEE	Total	Pre-requisite
2501AC01	Environmental Science	**FC**	2			0	100	-	100	**-**
2501AC02	Constitution of India	**FC**	2			0	100	-	100	**-**
2501AC03	Research Methodology	**FC**	2			0	100	-	100	**-**
2501AC04	Intellectual Property Rights & Patents	**FC**	2			0	100	-	100	**-**
2501AC05	Indian Knowledge Systems	**FC**	2			0	100	-	100	**-**
Total			**10**			**0**				

Environmental Science
(Common to CE, EEE, ME, ECE, CSE, IT, AIML, CSE(DS), PT & Min.E)

	L	T	P	C
Course Code: 2501AC01	**2**	**0**	**0**	**0**

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

CO1: Outline the natural resources and their importance for the sustenance of the life
CO2: Explain about the biodiversity of India, threats and its conservation methods
CO3: Illustrate various attributes of the pollution, impacts and measures to control the pollution along with waste management practices
CO4: Describe social issues of both rural and urban environment to combat the challenges and the legislations of India in environmental protection
CO5: Explain the population growth and its implications

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	1	-	-	-	1	-	2	-	-	-	-
CO2	1	-	-	-	-	-	3	-	-	-	-
CO3	-	-	-	-	2	-	3	-	-	-	1
CO4	-	-	-	-	1	-	3	1	-	-	1
CO5	-	-	-	-	-	-	3	-	-	-	-

UNIT-I

Multidisciplinary Nature of Environmental Studies: Definition, Scope and Importance, Need for Public Awareness.
Natural Resources: Renewable and non-renewable resources – Natural resources and associated problems.

UNIT-II

Ecosystem, Biodiversity and Its Conservation:
Ecosystems: Concept of an ecosystem. – Structure and function of an ecosystem – Producers, consumers, and decomposers. Food chains, food webs and ecological pyramids.
Biodiversity And Its Conservation: Definition: genetic, species and ecosystem diversity – Bio-geographical classification of India – Values of biodiversity. Hot-spots of biodiversity – Threats to biodiversity. Conservation of biodiversity: In-situ and Ex-situ conservation of biodiversity.

UNIT-III

Environmental Pollution and Solid Waste Management: Environmental Pollution: Definition, Cause, effects, and control measures of:
a. Air Pollution.
b. Water Pollution
c. Soil Pollution

d. Marine Pollution
e. Noise Pollution
Solid Waste Management:
Causes, effects and control measures of urban and industrial wastes – Role of an individual in prevention of pollution

UNIT-IV

Social Issues and The Environment: Social Issues and the Environment: From Unsustainable to Sustainable development – Urban problems related to Energy &Water. Resettlement and rehabilitation of people, Environmental ethics, Climate change, global warming, acid rain, ozone layer depletion, nuclear accidents, and holocaust. Environment Protection Act – Air (Prevention and Control of Pollution) Act. – Water (Prevention and control of Pollution) Act-Wildlife Protection Act – Forest Conservation Act – Issues involved in enforcement of environmental legislation – Public awareness.

UNIT–V

Human Population and The Environment: Population growth, variation among nations. Environment and human health, Human Rights, Value Education. Role of Information Technology in Environment and human health.

Text Books:

1. Textbook of Environmental Studies for Undergraduate Courses by Erach Bharucha for University Grants Commission, Universities Press, ISBN: 9788173718625
2. Environmental Studies by Palaniswamy – Pearson education, ISBN: 978-9332528277

Reference Books:

1. Textbook of Environmental Science, Deeksha Dave and E.Sai Baba Reddy, Cengage Publications. ISBN: 978-8131517604
2. Textbook of Environmental Sciences and Technology by M.Anji Reddy, B.S Publication. ISBN: 9385433385
3. Comprehensive Environmental studies by J.P.Sharma, Laxmi publications ISBN: 978-8170087380

Web Links:

1. https://archive.nptel.ac.in/courses/127/105/127105018/
2. https://onlinecourses.swayam2.ac.in/cec24_ge05/preview

Constitution of India

(Common to CE, EEE, ME, ECE, CSE, IT, AIML, CSE(DS), PT & Min.E)

	L	T	P	C
Course Code: 2501AC02	**2**	**0**	**0**	**0**

Course Outcomes:

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

- **CO1** Explain historical background of the constitution making and its importance for building a democratic India
- **CO2** Compare the functioning of three wings of the government i.e.., executive, legislative and judiciary
- **CO3** Interpret the value of the fundamental rights and duties for becoming good citizen of India
- **CO4** Compare the decentralization of power between central, state and local self-government
- **CO5** Extend the knowledge in strengthening of the constitutional institutions like CAG, Election Commission and UPSC for sustaining democracy.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	-	-	-	-	3	-	-	1	2	-	-
CO2	-	-	-	-	2	-	-	1	3	-	-
CO3	-	-	-	-	3	-	-	2	3	-	-
CO4	-	-	-	-	3	-	-	2	1	-	-
CO5	-	-	-	-	2	-	-	1	3	-	-

UNIT-I

Introduction to Indian Constitution: Constitution' meaning of the term, Indian Constitution - Sources and constitutional history, Features - Citizenship, Preamble, Fundamental Rights and Duties, Directive Principles of State Policy.

UNIT-II

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

UNIT-III

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

UNIT-IV

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

UNIT – V

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

Text Books:

1. Introduction to the Constitution of India, Durga Das Basu, Prentice – Hall of India Pvt. Ltd. New Delhi. ISBN: 978-9388548861
2. Indian Constitution, Subash Kashyap, National Book Trust. ISBN: 9788123707341

Reference Books:

1. Dynamics of Indian Government & Politics, J.A. Siwach ISBN: 9788120709768
2. Indian Government and Politics, D.C. Gupta ISBN: 9780706987782
3. Constitutional Law of India , H.M.Sreevai, 4th edition in 3 volumes (Universal Law Publication). ISBN: 9788194776529

Web Links:

1. nptel.ac.in/courses/109104045/
2. nptel.ac.in/courses/101104065/

Research Methodology

(Common to CE, EEE, ME, ECE, CSE, IT, AIML, CSE(DS), PT & Min.E)

	L	T	P	C
Course Code: 2501AC03	**2**	**0**	**0**	**0**

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

CO1: Explain the characteristics and process of research.
CO2: Select the research problem by applying problem identification techniques.
CO3: Formulate and execute research design process.
CO4: Report the results of research process adhering to professional ethics.
CO5: Analyze the results of research using statistical measures of central tendancy

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	1	-	-	2	-	-	-	-	-	2	1
CO2	1	-	-	2	-	-	-	-	-	2	1
CO3	1	-	-	2	-	-	-	-	-	2	1
CO4	1	-	-	2	-	-	-	3	-	2	1
CO5	1	-	-	2	2	-	-	-	-	2	1

Unit – I

Meaning of Research: Function of Research - Characteristics of Research – Steps involved in Research – Research in Pure and Applied Sciences - Inter Disciplinary Research. Factors which hinder Research – Significance of Research - Research and scientific methods – Research Process– Criteria of good Research – Problems encountered by Researchers – Literature review.

UNIT-II

Identification of Research Problem: Selecting the Research problem – Necessity of defining the problem – Goals and Criteria for identifying problems for research. Perception of Research problem – Techniques involved in defining the problem

UNIT-III

Research Design: Formulation of Research design – Need for Research design – Features of a good design – Important concepts related to Research design.

UNIT-IV

Interpretation and Report Writing: Meaning and Technique of interpretation – Precautions in interpretation – Significance of report writing – Different steps in writing a report – Layout of a Research report.

Unit – V

Statistical Techniques and Tools: Introduction of statistics – Functions – Limitations – Measures of central tendency - Arithmetic mean – Median – Mode – Standard deviation – Co-efficient of variation (Discrete serious and continuous serious) – Correlation - Regression

Text Books:

1. Research Methodology Methods & Techniques, C.R. Kothari – New Age international Publishers, Reprint. ISBN: 9788122415223
2. A Hand Book of Methodology of Research, Rajammall, P. Devadoss and K. Kulandaivel, RMM Vidyalaya press. ISBN: 9780367135720

Reference Books:

1. Thesis and Assignment Writing, J. Anderson, Wiley Eastern Ltd. ISBN: 978-8126530755
2. Research Methodology, Mukul Gupta, Deepa Gupta – PHI Learning Private Ltd., New Delhi. ISBN: 9788120349469
3. Fundamentals of Mathematical statistics, S.C. Gupta and V.K. Kapoor, Sultan Chand & Sons, New Delhi. ISBN: 978-8180545283

Web Links:

1. https://nptel.ac.in/courses/127106227
2. https://archive.nptel.ac.in/courses/121/106/121106007/

Intellectual Property Rights & Patents

(Common to CE, EEE, ME, ECE, CSE, IT, AIML, CSE(DS), PT & Min.E)

Course Code: 2501AC04	L	T	P	C
	2	0	0	0

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

CO1: Compare various types of Intellectual Property rights.
CO2: Discuss Intellectual Property and infer rights on such Intellectual Property owners
CO3: Explain the process of patenting
CO4: Apply for Trade marks and Copyrights.
CO5: Explain the methods to protect Trade secrets

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	1	-	-	-	-	1	-	3	-	-	2
CO2	1	-	-	-	-	1	-	3	-	-	2
CO3	1	-	-	-	-	1	-	3	-	-	2
CO4	1	-	-	-	-	1	-	3	-	-	2
CO5	1	-	-	-	-	1	-	3	-	-	2

UNIT-I

Introduction to Intellectual Property Rights (IPR): Concept of Property - Introduction to IPR – International Instruments and IPR - WIPO - TRIPS – WTO –Laws Relating to IPR - IPR Tool Kit - Protection and Regulation - Copyrights and Neighboring Rights – Industrial Property – Patents - Agencies for IPR Registration – Traditional Knowledge –Emerging Areas of IPR – Layout Designs and Integrated Circuits – Use and Misuse of Intellectual Property Rights.

UNIT-II

Copyrights and Neighboring Rights: Introduction to Copyrights – Principles of Copyright Protection – Law Relating to Copyrights - Subject Matters of Copyright – Copyright Ownership – Transfer and Duration – Right to Prepare Derivative Works –Rights of Distribution – Rights of Performers – Copyright Registration – Limitations – Infringement of Copyright – Relief and Remedy – Case Law - Semiconductor Chip Protection Act.

UNIT-III

Patents: Introduction to Patents - Laws Relating to Patents in India – Patent Requirements – Product Patent and Process Patent - Patent Search - Patent Registration and Granting of Patent - Exclusive Rights – Limitations - Ownership and Transfer — Revocation of Patent – Patent Appellate Board - Infringement of Patent – Double Patenting — Patent Cooperation Treaty – New developments in Patents – Software Protection and Computer related Innovations.

UNIT-IV

Trademarks and Trade secrets: Introduction to Trademarks – Laws Relating to Trademarks – Functions of Trademark – Distinction between Trademark and Property Mark – Marks Covered under Trademark Law - Trade Mark Registration – Trade Mark Maintenance – Transfer of rights - Deceptive Similarities - Likelihood of Confusion - Dilution of Ownership – Trademarks Claims and Infringement – Remedies – Passing off Action - Introduction to Trade Secrets – General Principles - Maintaining Trade Secret – Physical Security – Employee Access Limitation – Employee Confidentiality Agreements.

Unit – V

Cyber Law and Cyber Crime: Introduction to Cyber Law – Information Technology Act 2000 - Protection of Online and Computer Transactions - E-commerce - Data Security – Authentication and Confidentiality - Privacy - Digital Signatures – Certifying Authorities - Cyber Crimes - Prevention and Punishment – Liability of Network Providers. Relevant Cases Shall be dealt where ever necessary.

Text Books:

1. Fundamentals of IPR for Engineers- Kompal Bansal &Parishit Bansal, B. S. Publications (Press). ISBN: 9788178002774
2. Intellectual Property -Deborah E.Bouchoux, Third Edition, Cengage Learning, New Delhi. ISBN: 0 340 67786 4

Reference Books:

1. Intellectual property rights- Prabuddha Ganuli, Tata Mcgraw hill. ISBN: 978-0070077171
2. Intellectual property rights M.Ashok kumar and Mohd.Iqbal Ali:, Serials Publications ISBN: 9788183871648
3. Intellectual Property Rights (Patents & Cyber Law), Dr. A. Srinivas. Oxford University Press, New Delhi. ISBN: 979-8886848229
4. Intellectual Property- Richard Stim, Cengage Learning, New Delhi. ISBN: 978-0766826656

Web Links:

1. http://www.wipo.int/portal/en/index.html
2. https://indiankanoon.org/
3. http://www.ipindia.nic.in/patents.htm
4. http://www.ipindia.nic.in/trade-marks.htm
5. http://copyright.gov.in

Indian Knowledge Systems (IKS)

(CE, EEE, ME, ECE, CSE, IT, AIML, CSE(DS), PT, Min. E)

	L	T	P	C
Course Code: 2501AC05	**2**	**0**	**0**	**0**

Course Outcomes:

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

CO1: Describe the foundations and scope of Indian Knowledge Systems.
CO2: Recognize major Indian contributions to knowledge, science, and culture.
CO3: Appreciate ethical values and sustainability rooted in Indian traditions.
CO4: Relate traditional knowledge to contemporary societal and technological needs.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	-	-	-	-	-	3	1	1	-	-	-
CO2	-	-	-	-	-	3	1	1	-	-	-
CO3	-	-	-	-	-	3	1	1	-	-	-
CO4	-	-	-	-	-	3	1	1	-	-	-
CO5	-	-	-	-	-	3	1	1	-	-	-

UNIT – I
Foundations of Indian Knowledge Systems

Meaning, scope, and importance of IKS. Sources of IKS: Vedas, Upanishads, Itihasas, Puranas, Shastras. Traditional Knowledge, Indigenous Knowledge, and Western Knowledge – comparison. Gurukula system and knowledge transmission traditions

UNIT – II
Indian Civilization and Philosophical Traditions

Overview of ancient Indian civilization (Indus Valley, Maurya, Gupta). Indian philosophical schools and ethical values. Epics: Ramayana and Mahabharata – cultural and moral significance. Religious traditions: Hinduism, Buddhism, Jainism, Sikhism

UNIT – III
Arts, Architecture and Cultural Expressions

Temple architecture: Nagara, Dravidian, Vesara. Mughal architecture and monuments. Indian classical dance forms and music traditions. Handicrafts, sculpture, painting, and iconography

UNIT – IV
Traditional Knowledge in Health, Environment and Sustainability

Ayurveda and Yoga – holistic health systems. Traditional agriculture and food practices. Water conservation methods and environmental ethics. Biodiversity conservation and sustainable living

UNIT – V
Traditional Knowledge and Contemporary Relevance

Protection of Traditional Knowledge.Traditional Knowledge (TK) and Intellectual Property Rights (overview). Role of government and national initiatives. Relevance of IKS to engineering, innovation, and entrepreneurship

Text Books:

1. **Indian Knowledge Systems**, Kapil Kapoor & Avadhesh Kumar Singh, D.K. Printworld (India) Pvt. Ltd., New Delhi. ISBN: 9788124608177.
2. **Indian Philosophy – Volume I & II**, S. Radhakrishnan, Oxford University Press, New Delhi. ISBN: 9780195698418.
3. **Science and Technology in Ancient India**, O.P. Jaggi, Atma Ram & Sons, Delhi. ISBN: 9788185694085.
4 **Indian Heritage and Culture,** Madhukar K. Bhagat, McGraw Hill Education (India). ISBN: 9789332901476.
5 **Traditional Knowledge System in India,** Yogendra K. Malik, Pearson Education India. ISBN: 9788131762219.

Reference Books:

1. **Ancient Indian Science and Technology**, Debiprasad Chattopadhyaya, History of Science, Philosophy and Culture in Indian Civilization Series. ISBN: 9788187586098.
2. **Indian Culture and Heritage**, Nitin Singhania, McGraw Hill Education. ISBN: 9789353161132.
3. **Knowledge Traditions of India**, Bhu Dev Sharma, Indian Council of Philosophical Research (ICPR). ISBN: 9788185636306.

Web Links:

1. https://iks.aicte-india.org/
2. https://www.education.gov.in/iks

Minor Stream: Analytics and Security (In Collaboration with Microsoft)

Course Code	Course Name	Level	L	T	P	C	CIE	SEE	Total	Pre-requisite
2501AI36	Network Security Essentials	FC	2			2	50	50	100	-
2501AI37	Introduction to System Design	IC	2		1	3	50	50	100	-
2501AI38	Introduction to Generative AI	IC	2		1	3	50	50	100	-
2501AI39	Cloud Security Essentials	IC	2		1	3	50	50	100	-
2501AI40	Cyber Security Essentials	IC	2		1	3	50	50	100	-
2501AI41	Fundamentals of Block Chain	**IC**	2		1	3	50	50	100	-
2501AI42	AI for Cyber Threat Intelligence	IC	2		1	3	50	50	100	-
2501AI43	Generative Adversarial Networks Fundamentals	AC	2		1	3	50	50	100	-
2501AI44	AI in Healthcare Security	AC	2		1	3	50	50	100	-
2501AI45	Advanced Topics in Generative AI	AC	2		1	3	50	50	100	-
2501AI46	Ethical AI and Responsible Computing	AC	2		1	3	50	50	100	-
Total			22		**10**	**32**				

Network Security Essentials

Course Code:2501AI36	L	T	P	C
	2	0	0	2

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

CO1: Explain fundamental concepts of network security and the need for secure communication.
CO2: Analyze the security aspects of common network protocols and apply standards.
CO3: Configure and use firewalls, VPNs, and IDPS for network protection.
CO4: Implement secure remote access, wireless security, and best practices.
CO5: Use tools for network security monitoring and mitigate DoS/DDoS attacks.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	3	2	1	1	1	-	-	-	2	-	2
CO2	3	3	2	2	2	-	-	-	2	-	3
CO3	2	3	3	2	3	-	-	-	2	1	3
CO4	2	2	2	2	3	-	-	-	2	2	3
CO5	2	3	3	2	3	-	-	-	2	1	3

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1	-	2
CO2	-	2
CO3	-	2
CO4	-	2
CO5	-	2

UNIT-I

Introduction to Network Security: Introduction to network security and types of threats, CIA triad and security protocols, firewalls and IDS/IPS systems, antivirus and anti-malware solutions, VPNs and wireless security, access control models and encryption/cryptography, security policies and threat landscape, compliance and regulations, impact of breaches, and remote work with cloud security

UNIT-II

Protocols and Security Standards
TCP/IP stack and DNS resolution, HTTP methods and HTTPS encryption, Packet sniffing and IP spoofing, DNS cache poisoning and man-in-the-middle attacks, TLS/SSL protocols and secure protocol design principles, Firewall, IDS/IPS, and network access control policies, Port scanning tools (e.g., Nmap) and zero trust architecture, Network security best practices

UNIT-III

Firewalls, VPNs, and IDPS: Stateful vs stateless firewalls and packet filtering, Next-generation firewall (NGFW) and firewall rule configuration, IPSec tunneling protocols and SSL/TLS VPNs, IDS vs IPS and detection methods (signature-based & anomaly-based), Deep packet inspection and zero-day attack detection, VPN encryption protocols and endpoint threat monitoring, Security incident response and network segmentation, Threat monitoring and prevention strategies

UNIT-IV

Wireless and Remote Security: Wi-Fi encryption and rogue access point detection, Evil twin attacks and WIDS, VPN and SSH secure tunneling, SSH key management and brute-force prevention, 2FA/MFA authentication methods, Secure remote access and zero trust, MITM attack prevention and remote desktop security, Endpoint protection and remote endpoint security

UNIT – V

Monitoring, DoS/DDoS & Best Practices:
Network traffic analysis and packet inspection, Intrusion detection and anomaly-based threat detection, Log correlation and SIEM tools, DoS/DDoS mitigation and traffic filtering, Application layer and behavioral attack detection, Access control and least privilege policies, Defense in depth and real-time alerting, Honeypots and deception technologies.

Text Books:

1 Information Security Fundamentals by IBM ICE Publications.
2 Cryptography and Network Security, William Stallings, Pearson Education, 411i Edition
3 Cryptography and Network Security, Atul Kahate, McGrawHill, 2nd Edition

Reference Books:

1 Introduction to Network Security, Neal Krawetz, CENGAGE Learning
2 Information Security, Principles and Practice, Mark Stamp, Wiley India.

Web Links:

1 https://www.geeksforgeeks.org/computer-networks/network-security/
2 https://learn.microsoft.com/en-us/azure/security/fundamentals/network-overview

Introduction to System Design

Course Code: 2501AI37	L	T	P	C
	2	0	1	3

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

CO1: Explain system design fundamentals and analyze system requirements.
CO2: Apply core design principles and assess modularity, abstraction, and coupling.
CO3: Compare architectural styles and apply design patterns for system development.
CO4: Design scalable, high-performance systems with optimized data modeling.
CO5: Incorporate security, fault tolerance, and reliability in large-scale system designs.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	3	3	2	2	1	-	-	-	-	2	2
CO2	3	3	3	1	2	-	-	-	-	1	2
CO3	3	2	3	1	2	-	-	-	-	1	2
CO4	3	3	3	2	3	-	-	-	-	2	2
CO5	3	3	3	2	2	-	-	-	-	2	2

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1	-	2
CO2	-	2
CO3	-	2
CO4	-	2
CO5	-	2

UNIT-I

System Design Basics & Requirements: System design fundamentals, software architecture patterns, scalability and load balancing, functional and non-functional requirements (NFRs), high availability and fault tolerance, data consistency models and caching strategies, microservices architecture, CAP theorem, security in system design, performance analysis and design for failure

Practice:
1. Define functional and non-functional requirements for an e-commerce system
2. Draw a context-level diagram and requirement document

UNIT-II

Design Principles & Patterns: Object-oriented principles, abstraction in software design, modularity and reusability, encapsulation and information hiding, low coupling and high cohesion, SOLID principles, design pattern catalog (GoF) including Singleton, Factory, Observer, and Strategy patterns, threat modeling in design, secure design principles, anti-patterns and code smells, and design policy enforcement tools

Practice:

1. Implement a modular system using abstraction and encapsulation
2. Apply Singleton and Factory design patterns to a basic problem

UNIT-III

Architectural Styles: Client-server model, layered architecture pattern, monolithic and microservices architecture, service decomposition and inter-service communication (REST/gRPC), scalability and performance trade-offs, deployment complexity and fault isolation, API gateway pattern and load balancing strategies, containerization (Docker/Kubernetes), security risks in distributed systems, service discovery mechanisms, and architecture decision records (ADR)

Practice:

1. Compare microservices and monolithic architectures
2. Design a client-server or layered architecture diagram

UNIT-IV

Database Design & Performance: Database normalization forms, indexing strategies (B-Tree, Hash), entity-relationship modeling and query execution plans, SQL query optimization techniques, read/write load balancing and sharding (horizontal/vertical), in-memory caching (Redis/Memcached) and caching invalidation policies, database partitioning and denormalization trade-offs, data consistency and replication, SQL injection prevention, database performance monitoring tools, and CAP theorem in distributed databases

Practice:

1. Design a normalized schema and ER diagram for a system
2. Analyze index and query optimization for given queries

UNIT – V

Security, Reliability & Case Studies: Security hardening techniques and patch management policies, principle of least privilege with threat modeling and risk assessment, backup and recovery strategies, high availability architecture with redundancy and failover systems, disaster recovery planning with RTO and RPO metrics, SIEM and incident response planning, real-world breach case studies, business continuity planning, cloud-based disaster recovery, and compliance and regulatory requirements (e.g., GDPR, ISO 27001).

Practice:

1. Evaluate a system design for potential security flaws

2. Present trade-offs and analysis from a real-world system case study

Text Books:

1 A. Xu, System Design Interview – An Insider's Guide, ByteByteGo, 2020.
2 M. Kleppmann, Designing Data-Intensive Applications, O'Reilly Media, 2017.
3 M. Richards and N. Ford, Fundamentals of Software Architecture, O'Reilly Media, 2020.

Reference Books:

1 N. Ford and M. Richards, Software Architecture: The Hard Parts, O'Reilly Media, 2021.
2 R. C. Martin, Clean Architecture: A Craftsman's Guide to Software Structure and Design, Pearson, 2017.
3 M. L. Abbott and M. T. Fisher, *The Art of Scalability*, Addison-Wesley, 2015.
4 C. Richardson, Microservices Patterns, Manning Publications, 2018

Web Links:

1 https://www.geeksforgeeks.org/system-design/system-design-tutorial/
2 https://www.ibm.com/docs/en/ram/7.5.4?topic=management-system-design
3 https://nptel.ac.in/courses/107106009

Introduction to Generative AI

	L	T	P	C
Course Code:2501AI38	**2**	**0**	**1**	**3**

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

CO1: Explain the foundational concepts and types of generative AI models.
CO2: Use Python and libraries like TensorFlow/PyTorch for generative model implementation.
CO3: Prepare and preprocess datasets for effective generative model training.
CO4: Build and evaluate GANs, VAEs, and sequence generation models.
CO5: Apply generative AI to real-world problems and analyze model outcomes.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	3	2	-	-	2	-	-	-	-	-	3
CO2	3	2	2	1	3	-	-	-	-	-	2
CO3	2	3	2	2	3	-	-	-	-	-	2
CO4	3	3	3	2	3	-	-	-	-	-	2
CO5	2	3	3	3	3	-	-	-	2	1	3

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1	-	2
CO2	-	2
CO3	-	3
CO4	-	2
CO5	-	2

UNIT-I

Introduction to Generative AI: Generative model fundamentals, GAN architecture and training basics, Variational Autoencoders (VAEs), Autoencoders and representation learning, Sequence generation models (LSTM, GRU), Python syntax and data structures, TensorFlow and PyTorch workflows, Secure and ethical coding practices

Practice:
1. Implement a basic image generator using NumPy
2. Use PyTorch to build a simple autoencoder for MNIST

UNIT-II

Data Preprocessing for Generative Tasks: Data preprocessing and cleaning techniques, Data deduplication and normalization, Data augmentation strategies, Bias and noise mitigation, Privacy-preserving data processing, Tools for dataset preparation, Dependency and environment management, Ethical data governance and compliance

Practice:
1. Preprocess and augment image data for GANs
2. Preprocess and augment image data for GANs

UNIT-III

Implementation of Generative Models: Generator vs Discriminator roles, Adversarial training loop, Loss functions: Binary Cross, Entropy, Wasserstein, Mode collapse and training challenges, VAE architecture and latent space modeling, Reparameterization trick, Autoencoders: denoising, feature extraction- Implementation using TensorFlow and PyTorch

Practice:

1. Implement a VAE to reconstruct and generate digit images
2. Create a denoising autoencoder and evaluate performance

UNIT-IV

Advanced Generative Architectures: Transformer architecture and pretraining principles, Attention mechanisms and prompt engineering, Language generation and protocol tuning, Overfitting and regularization threats, Bias and fairness in language models, Real-world generative use cases: images, text, healthcare

Practice:

1. Train an LSTM model for character-level text generation
2. Evaluate generative models using FID and Inception Score

UNIT – V

Evaluation, Applications & Ethical Considerations: Inception Score (IS) and FID metrics, Output quality evaluation and benchmarking, Deepfake detection and adversarial risks, Regulatory and compliance policies, Generative AI misuse and ethical risks, Content authenticity and governance, Industry applications and real-world relevance

Practice:

1. Use a pre-trained GPT model for text generation and fine-tuning
2. Develop a mini-project applying generative AI in a selected domain

TextBooks:

1 Ian Goodfellow, Yoshua Bengio, Aaron Courville, *Deep Learning*, MIT Press
2 David Foster, *Generative Deep Learning: Teaching Machines to Paint, Write, Compose, and Play*, O'Reilly
3 Francois Chollet, *Deep Learning with Python*, Manning Publications

Reference Books:

1 PyTorch and TensorFlow official documentation
2 Tutorials from DeepLearning.ai and Coursera on GANs and VAEs
3 Research papers on generative models (NIPS, ICLR, CVPR)

Web Links:

1 https://www.cloudskillsboost.google/course_templates/536
2 https://www.coursera.org/learn/introduction-to-generative-ai

Cloud Security Essentials

	L	T	P	C
Course Code:2501AI39	**2**	**0**	**1**	**3**

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

CO1: Describe cloud computing concepts and assess cloud-specific security challenges.
CO2: Apply identity and access management strategies including MFA and RBAC.

CO3: Implement encryption, key management, and network security controls in cloud platforms.
CO4: Monitor cloud security events, perform logging, and handle security incidents.
CO5: Evaluate compliance needs and implement governance and cloud security best practices.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	3	2	2	2	1	1	2	2	-	-	1
CO2	3	3	3	2	2	1	2	2	-	-	1
CO3	3	3	3	2	3	2	1	2	-	-	2
CO4	3	2	3	2	3	2	2	2	-	-	2
CO5	2	2	2	2	2	3	3	3	-	-	3

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1	-	2
CO2	-	2
CO3	-	1
CO4	-	1
CO5	-	2

UNIT-I

Cloud Security Fundamentals: Cloud service and deployment models, shared responsibility model, identity and access management, cloud access security brokers, data encryption and protection, misconfiguration and API security risks, cloud-native and workload protection tools, compliance frameworks and incident response

Practice:

1. Create a free-tier cloud account and explore services
2. Identify and document shared responsibility for selected services

UNIT-II

IAM and Access Security: Principle of least privilege, Single sign-on (SSO) and identity providers (IdPs), Role hierarchy and permission sets, Authentication vs authorization, MFA methods and credential protection, OAuth 2.0 and OpenID Connect, Privileged access management (PAM), IAM policies and enforcement

Practice:

1. Create IAM users, roles, and groups in a cloud platform
2. Enable and test MFA for cloud access

UNIT-III

Encryption, Key and Network Security: Encryption techniques and types (AES, RSA, symmetric, asymmetric), Key Management Services (KMS) and key lifecycle, Envelope

encryption and HSM, Virtual Private Cloud (VPC) security, Security Groups and NACLs, Firewall types and configurations, VPN and TLS/SSL protocols, Data Leakage Prevention (DLP)

Practice:

1. Encrypt cloud storage with a customer-managed key
2. Set up a secure VPC with firewall and subnet controls

UNIT-IV

Monitoring, Logging and Incident Response: Security logging practices and retention, Log analysis and SIEM integration, Real-time threat detection and response, Incident response lifecycle and automation, Forensic investigation techniques, Security event normalization, Alert tuning and prioritization, Compliance and audit logging

Practice:

1. Enable logging and monitor events in cloud console
2. Simulate and document a simple incident response scenario

UNIT – V

Compliance and Best Practices: Data classification and protection, Compliance frameworks (GDPR,HIPAA,ISO), Cloud governance models, Cloud Security Posture Management (CSPM), Data residency and sovereignty, Encryption and tokenization, Policy enforcement and automation, Risk management and assessment, Data Loss Prevention (DLP), Security best practices in cloud environments.

Practice:

1. Run a compliance check using built-in cloud tools
2. Create a checklist of cloud security best practices

TextBooks:

1 K. Hashizume et al., *An Analysis of Cloud Computing Security Issues*, Springer, 2021
2 G. Reese, *Cloud Application Architectures*, O'Reilly Media, 2009.
3 T. Krutz and R. Dean, *Cloud Security: A Comprehensive Guide to Secure Cloud Computing*, Wiley, 2010

Reference Books:

1 Rittinghouse and J. Ransome, Cloud Computing: Implementation, Management, and Security, CRC Press, 2016
2 B. Grobauer, T. Walloschek, and E. Stöcker, Understanding Cloud Computing Vulnerabilities, IEEE Security & Privacy, 2011
3 D. Linthicum, Cloud Security Basics, O'Reilly Media, 2022

Web Links:

1 https://www.geeksforgeeks.org/software-engineering/cloud-computing-security/
2 https://www.ibm.com/think/topics/cloud-security

Cyber Security Essentials

Course Code:2501AI40	L	T	P	C
	2	0	1	3

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

CO1: Explain core cybersecurity concepts, threat actors, and risk assessment strategies.
CO2: Analyze cyber threats and plan incident responses using appropriate tools.
CO3: Apply cryptographic techniques and secure protocols to protect data in transit and at rest.
CO4: Configure and test technical controls such as firewalls, VPNs, and security infrastructure.
CO5: Implement access control models, conduct penetration testing, and ensure compliance with cybersecurity regulations

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	3	2	2	2	1	-	-	-	-	-	1
CO2	3	3	3	3	2	-	-	-	1	1	2
CO3	3	3	3	2	2	-	-	-	-	1	2
CO4	3	3	3	3	3	-	-	-	1	2	2
CO5	3	3	3	3	3	-	-	-	2	2	3

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1	-	2
CO2	-	1
CO3	-	1
CO4	-	2
CO5	-	2

UNIT-I

Foundations of Cybersecurity: CIA Triad and core security principles, Access control and least privilege enforcement, Defense in depth and layered security, Threat actors and malware classification, Social engineering and human factors, Risk assessment and incident response, Security tools and endpoint protection, Cybersecurity frameworks and awareness training

Practice:
1. Research recent cyber attacks and present threat types,
2. Simulate a phishing or fake login page for awareness

UNIT-II

Web Security and Cryptography: OWASP Top 10 vulnerabilities, Injection attacks (SQL, Command), Broken authentication, Sensitive data exposure, HTTPS and TLS

protocols, TLS handshake process, Symmetric encryption (AES), Asymmetric encryption (RSA), Hashing algorithms (SHA-256), Digital signatures and integrity, Public Key Infrastructure (PKI), SSL/TLS certificate validation, Secure key management, Cryptographic attack vectors (brute force, padding oracle), Secure coding best practices

Practice:

1. Perform OWASP web vulnerability testing using Juice Shop or Burp Suite,
2. Capture and analyze HTTPS traffic using Wireshark.

UNIT-III

Access Control and IAM: Discretionary Access Control (DAC), Role-Based Access Control (RBAC), Attribute-Based Access Control (ABAC), Identity and Access Management (IAM), Multi-Factor Authentication (MFA), Single Sign-On (SSO), Authentication vs Authorization, OAuth 2.0 protocol, OpenID Connect, Identity Federation, Credential theft prevention, Privileged Access Management (PAM), Identity lifecycle management, Access control policies, Insider threat mitigation

Practice:

1. Configure MFA and role-based access control in a demo web app,
2. Set up and manage user identities using OpenLDAP

UNIT-IV

Technical Controls and Infrastructure: Network intrusion detection systems (NIDS), Network intrusion prevention systems (NIPS), Virtual Private Networks (VPN) protocols (IPSec, SSL/TLS), Firewall rule configuration, Next-generation firewalls (NGFW), Deep packet inspection (DPI), Traffic monitoring tools (Wireshark, Zeek), Port scanning and enumeration (Nmap), Secure tunneling protocols, Access control lists (ACLs), Bandwidth and anomaly analysis, Zero Trust Network Architecture, Threat intelligence integration, Security logging and alerting, Network segmentation and isolation

Practice:

1. Set up a basic firewall using UFW or pfSense
2. Configure a VPN connection and verify encryption

UNIT – V

Compliance, Ethics, and Auditing: General Data Protection Regulation (GDPR), Health Insurance Portability and Accountability Act (HIPAA), Payment Card Industry Data Security Standard (PCI DSS), Data privacy principles and data minimization, Digital rights and ethical hacking, Security compliance frameworks and continuous compliance monitoring, Risk assessment, mitigation and third-party risk management, Security audit processes, governance and legal implications of cyber threats and breach notification policies, Privacy Impact Assessments (PIA)

Practice:

1. Conduct a mock compliance audit (GDPR, HIPAA, etc.)
2. Draft a simple security policy for a small organization

Text Books:

1 W. Stallings, Network Security Essentials: Applications and Standards, Pearson, 2016
2 M. Whitman and H. Mattord, Principles of Information Security, Cengage Learning, 2021
3 C. P. Pfleeger and S. L. Pfleeger, Security in Computing, Pearson, 2015

Reference Books:

1 M. Sikorski and A. Honig, Practical Malware Analysis, No Starch Press, 2012
2 S. Malik, Security Operations Center: Building, Operating, and Maintaining Your SOC, Apress, 2020
3 T. Holt et al., Cybercrime and Digital Forensics: An Introduction, Routledge, 2018
4 R. Bejtlich, The Practice of Network Security Monitoring, No Starch Press, 2013

Web Links:

1 https://www.coursera.org/learn/cyber-security-fundamentals
2 https://learn.microsoft.com/en-us/training/paths/describe-basic-concepts-of-cybersecurity/

Fundamentals of Block Chain

Course Code:2501AI41	L	T	P	C
	2	0	1	3

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

CO1: Explain the foundational concepts of blockchain and distributed systems
CO2: Demonstrate cryptographic techniques and consensus mechanisms used in blockchain
CO3: Develop smart contracts and deploy decentralized applications
CO4: Identify real-world blockchain use cases and evaluate their feasibility
CO5: Analyze blockchain limitations, trends, and regulatory concerns

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	3	2	-	-	2	-	-	-	-	-	3
CO2	3	3	2	2	3	-	-	-	-	-	2
CO3	3	2	3	2	3	-	-	-	2	2	2
CO4	2	3	2	2	3	-	-	-	2	2	3
CO5	2	3	2	2	2	-	-	-	3	2	3

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1	-	2
CO2	-	2
CO3	-	2
CO4	-	3
CO5	-	3

UNIT-I

Introduction to Blockchain: Consensus algorithms, Blockchain models, Cryptographic hashing, Block creation and validation, Node synchronization, Smart contracts, Scalability and security in decentralized systems, Blockchain applications and compliance

Practice:
1. Simulate a basic blockchain with chained blocks and hash computation
2. Implement SHA-256 and verify hash changes with block data

UNIT-II

Cryptographic Foundations: Symmetric and asymmetric cryptography, Cryptographic hash functions, Digital signatures and PKI, Key management and trust models, Cryptanalysis and security threats, Consensus protocols (PoW vs PoS), Data confidentiality and integrity, Quantum-safe and regulatory-compliant cryptography

Practice:

1. Demonstrate digital signatures and key pair generation using Python
2. Set up a local Ethereum blockchain using Geth

UNIT-III

Smart Contracts and Platforms: Ethereum architecture and EVM, Solidity programming and contract deployment, Gas optimization and DApp design, Smart contract security and tools (Remix, Truffle), Blockchain oracles and integrations, Hyperledger Fabric and chaincode, Consensus mechanisms and access control (MSP), Enterprise blockchain scalability and interoperability

Practice:

1. Develop and deploy a simple smart contract using Solidity in Remix
2. Connect a smart contract to frontend using Web3.js and MetaMask

UNIT-IV

Blockchain Use Cases: DeFi and cross-border payments, Blockchain in supply chain and logistics, Healthcare privacy and EHR systems, Secure voting and digital identity (SSI), IoT authentication and data integrity, NFT standards, ownership, and provenance, Regulatory compliance and cybersecurity risks, Cross-platform blockchain interoperability

Practice:

1. Explore real-world blockchain use case – land registry or supply chain
2. Deploy an ERC-20 token or NFT (ERC-721) on testnet

UNIT – V

Challenges & Future Trends: Layer 2 scaling solutions, blockchain interoperability, regulatory compliance and legal frameworks, smart contract security, Web3 identity and access control, DeFi risks and vulnerabilities, protocol optimization and composability, tokenomics and governance

Practice:

1. Analyze transaction data and gas fees on Ethereum testnet
2. Present current trends in blockchain (e.g., Web3, Layer 2, DeFi, ZK-rollups)

TextBooks:

1 Imran Bashir, *Mastering Blockchain*, Packt Publishing
2 Melanie Swan, *Blockchain: Blueprint for a New Economy*, O'Reilly Media
3 Andreas M. Antonopoulos, *Mastering Ethereum*, O'Reilly Media

Reference Books:

1 Arvind Narayanan et al., *Bitcoin and Cryptocurrency Technologies*, Princeton University Press
2 Joseph Bonneau et al., *SoK: Research Perspectives and Challenges for Bitcoin and Cryptocurrencies* (IEEE)

3 Hyperledger Fabric documentation – https://hyperledger-fabric.readthedocs.io/

Web Links:

1 https://learn.microsoft.com/en-us/shows/beginners-series-to-blockchain/
2 https://learn.microsoft.com/en-us/archive/msdn-magazine/2018/march/blockchain-blockchain-fundamentals

AI for Cyber Threat Intelligence

	L	T	P	C
Course Code:2501AI42	**2**	**0**	**1**	**3**

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

CO1: Describe AI concepts and their application to cyber threat intelligence
CO2: Build ML models for threat detection, anomaly detection, and alert classification
CO3: Apply NLP and deep learning for malware and phishing analysis
CO4: Automate CTI tasks such as incident response and SOC operations using AI
CO5: Analyze limitations, adversarial threats, and ethics in AI-driven security systems

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	3	2	1	2	-	-	-	-	-	-	2
CO2	3	3	3	2	1	-	-	-	-	1	2
CO3	3	3	3	2	1	-	-	-	-	-	2
CO4	3	2	3	2	-	-	-	-	-	2	2
CO5	2	2	2	2	-	-	-	-	-	-	2

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1	-	2
CO2	-	2
CO3	-	2
CO4	-	2
CO5	-	2

UNIT-I

Introduction to AI in Cybersecurity: Cyber Threat Intelligence (CTI) lifecycle, threat intelligence types (tactical, operational, strategic), threat actor profiling and TTPs, indicators of compromise (IoCs), threat modeling and risk assessment, AI/ML in threat detection, threat data enrichment and correlation, threat intelligence platforms and protocols (STIX, TAXII), threat hunting and incident response integration, automation and policy-based mitigation

Practice:
1. Explore threat intelligence feeds and IoC extraction

UNIT-II

ML Techniques for Threat Detection: Supervised and unsupervised learning for threat detection, clustering and classification models (K-Means, DBSCAN, SVM, Random Forest), feature extraction and selection in cybersecurity, DDoS and insider threat detection

using ML, time-series analysis for threat detection, reducing false positives and evaluating models (precision, recall, F1), real-time intrusion detection with ML, adversarial machine learning threats, data labeling challenges, and integration of ML models into SIEM and security policy enforcement

Practice:

1. Build a supervised ML model to detect phishing URLs
2. Perform anomaly detection using clustering on network logs

UNIT-III

NLP and Deep Learning in Threat Intelligence: Phishing detection using NLP, Malware behavior analysis via text mining, Log parsing and normalization, Threat intelligence report mining, Named entity recognition in cybersecurity texts, Word embeddings for threat classification, RNNs for sequential log analysis, CNNs for network traffic classification, Transformer models in cybersecurity, Semantic analysis of threat reports, Protocol anomaly detection using deep learning, NLP-based spam and phishing email filtering, Tokenization and feature extraction from logs, Policy compliance through log analysis, Cyber threat hunting using deep neural networks

Practice:

1. Use NLP to extract threats from CTI reports
2. Classify malware samples using CNN models

UNIT-IV

Automated Threat Intelligence and SOC Integration: SIEM architecture and data pipelines, Real-time threat detection and correlation, Alert fatigue and triage optimization, Risk-based case prioritization, Playbook automation for incident response, MITRE ATT&CK integration with SIEM, Threat intelligence enrichment in alerts, Anomaly detection in security events, Reinforcement learningfor adaptive defense, Autonomous incident response systems, Policy-driven alert classification, Behavioral analytics for SOC operations, Protocol-aware event normalization, Threat scoring using machine learning, Modern SOAR-SIEM integration

Practice:

1. Automate alert triage using AI on SIEM data
2. Simulate incident response workflow with intelligent decision trees

UNIT – V

Challenges, Ethics, and Adversarial AI: Adversarial example generation techniques, Model poisoning and data integrity threats, Evasion attacks in deployed ML models, Explainable AI for security decision-making, Differential privacy in model training, Secure federated learning protocols, AI model auditing and risk assessment, Robustness evaluation

metrics for ML models, Regulatory frameworks for AI security (e.g., EU AI Act), Privacy-preserving machine learning, Secure multi-party computation in AI, Threat modeling for ML systems, Policy compliance in AI model deployment, Detection and mitigation of AI supply chain attacks, Interpretable ML tools for SOC analysts

Practice:

1. Evaluate model robustness against adversarial attacks
2. Use explainable AI (XAI) to interpret threat model decisions
3. Present a case study on real-world AI threat intelligence system

Text Books:

1 Sumeet Dua and Xian Du, *Data Mining and Machine Learning in Cybersecurity*, CRC Press
2 Clarence Chio and David Freeman, *Machine Learning and Security*, O'Reilly Media
3 Nina Godbole and Sunit Belapure, *Cybersecurity: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives*, Wiley

Reference Books:

1 MITRE ATT&CK Framework – https://attack.mitre.org/
2 IBM X-Force Exchange – https://exchange.xforce.ibmcloud.com/
3 Research papers on adversarial ML in cybersecurity (IEEE/ACM)

Web Links:

1 https://learn.microsoft.com/en-us/credentials/certifications/cybersecurity-architect-expert/

Generative Adversarial Networks Fundamentals

	L	T	P	C
Course Code:2501AI43	**2**	**0**	**1**	**3**

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

CO1: Explain the architecture, components, and purpose of GANs.
CO2: Implement various types of GANs using TensorFlow or PyTorch
CO3: Apply GANs to tasks like image generation, style transfer, and image translation.
CO4: Evaluate advanced GAN techniques and their applications
CO5: Identify ethical implications and limitations associated with GANs.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	3	2	2	2	1	1	-	1	-	-	1
CO2	3	3	3	3	3	1	-	-	-	-	2
CO3	3	3	3	3	3	1	1	-	-	-	2
CO4	3	3	3	3	2	2	1	-	-	-	2
CO5	2	2	2	1	1	3	2	3	1	-	2

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1	-	2
CO2	-	1
CO3	-	1
CO4	-	2
CO5	-	2

UNIT-I

Foundations of GANs: Generative Adversarial Networks (GANs), Generator vs Discriminator and adversarial training, GAN loss functions and mode collapse, DCGAN and Conditional GANs (cGAN), CycleGAN and StyleGAN architectures, TensorFlow and PyTorch GAN implementations, Dataset preparation and evaluation metrics (FID, IS), Ethical concerns, security threats and responsible AI guidelines

Practice:
1. Train a Deep Convolutional GAN (DCGAN) to generate new images and check its quality.
2. Generate class-specific images with cGAN and try unpaired image translation using CycleGAN.

UNIT-II

Basic GANs and Conditional GANs: Basic GAN implementation and architecture design, Adversarial loss functions and training stability, Conditional GANs and label conditioning, Dataset labeling and preprocessing techniques, Mode collapse prevention and generalization, GAN training with TensorFlow/PyTorch, Evaluation metrics (FID, IS) and output analysis, Use cases of cGANs (image-to-image, data augmentation), Ethical concerns, deepfake detection, and content regulation policies

Practice:

1. Build a simple GAN to generate images, watch training behavior, and try one fix for mode collapse.

UNIT-III

DCGANs, CycleGANs and PGGANs: DCGAN design and convolutional architecture, Normalization and activation strategies, CycleGAN for unpaired image translation, Generator consistency and domain transfer, Progressive GANs and resolution scaling, Training stability and GAN frameworks, Synthetic data risks and authenticity challenges, Ethical AI use and policy development.

Practice:

1. Build a DCGAN to generate images, learn normalization/activation choices, then try a simple progressive upscaling (PGGAN-lite) to improve resolution

UNIT-IV

Style GANs and Applications: StyleGAN architecture and improvements, Latent space and style manipulation, GANs for data augmentation and imbalance handling, Anomaly detection using GANs, GAN frameworks (PyTorch/TensorFlow), Synthetic data risks and adversarial robustness, Deepfake detection and misuse prevention, Ethical governance and explainability of GAN outputs

Practice:

1. Generate a few images using a pretrained StyleGAN generator, perform a small latent interpolation, and discuss possible applications, risks, and ethics.

UNIT – V

Ethics and Advanced Techniques: Deepfake generation and detection, Privacy and ethical risks of synthetic data, GAN training challenges (convergence issues, mode collapse, evaluation), Advanced GAN architectures (WGAN-GP, SAGAN, BigGAN), AI-generated content regulations, Bias and fairness in GAN outputs, Content authenticity and verification, Governance and responsible AI frameworks

Practice:

1. Learn how deepfakes are detected, why they're risky, and how policy/ethics guide safe work.
2. Implement a small WGAN-GP (or adapt a given tiny implementation), observe training stability and mode collapse, and test for bias in generated outputs. Propose governance / mitigation steps.

Text Books:

1 Goodfellow et al., Deep Learning, MIT Press, 2016

2 A. Rosebrock, Deep Learning for Computer Vision with Python, PyImageSearch, 2019
3 I. Goodfellow, Y. Bengio, and A. Courville, Generative Adversarial Networks, MIT Deep Learning Series, 2020

Reference Books:

1 J. Brownlee, Generative Adversarial Networks with Python, Machine Learning Mastery, 2021
2 S. Li and L. Wand, Progressive Growing of GANs for Improved Image Synthesis, arXiv preprint, 2018
3 A. Brock et al., Large Scale GAN Training for High Fidelity Natural Image Synthesis, ICLR, 2019
4 H. Zhang et al., Self-Attention Generative Adversarial Networks, ICML, 2019

Web Links:

1 https://developers.google.com/machine-learning/gan/gan_structure
2 https://www.geeksforgeeks.org/deep-learning/generative-adversarial-network-gan/

AI in Healthcare Security

	L	T	P	C
Course Code:2501AI44	**2**	**0**	**1**	**3**

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

CO1: Explain the role of AI in healthcare security and data protection
CO2: Analyze healthcare data security standards and regulatory frameworks.
CO3: Apply AI methods like ML, NLP, and generative models to detect and prevent threats

CO4: Design privacy-preserving AI systems for healthcare environments

CO5: Evaluate healthcare security incidents and propose AI-driven compliance solutions.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	3	2	2	2	2	2	-	-	-	-	2
CO2	3	3	3	3	3	1	-	-	-	-	3
CO3	3	3	3	2	2	3	-	-	-	1	2
CO4	3	2	2	2	1	3	-	-	-	1	2
CO5	2	2	2	1	1	2	-	-	-	2	2

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1	-	2
CO2	-	1
CO3	-	1
CO4	-	2
CO5	-	2

UNIT-I

Introduction to AI in Healthcare Security: AI-driven threat detection, data privacy and HIPAA compliance, anomaly detection in patient data, encryption and access control protocols, Zero Trust architecture in healthcare, cybersecurity policies and governance, ransomware and phishing threats, privacy-preserving machine learning, real-time monitoring and alerting tools

Practice:

1. Analyze anonymized healthcare datasets and identify data privacy issues

UNIT-II

Healthcare Data Security and Regulatory Frameworks: HIPAA Security Rule, PHI encryption protocols, access control and authentication, data breach threats, compliance monitoring tools, risk management frameworks, regulatory enforcement and penalties, cloud and mobile security challenges, privacy policies and audit requirements

Practice:

1. Implement HIPAA-compliant data handling procedures in AI workflows

UNIT-III

Core AI Techniques in Healthcare Security: Threat intelligence platforms, anomaly detection algorithms, behavioral analysis, zero-day threat response, data encryption standards (AES, RSA), key management protocols, intrusion detection systems (IDS),

cloud data protection, AI-driven threat identification

Practice:

1. Train a supervised ML model for malware detection in healthcare
2. Use NLP to detect phishing or fraud in medical communication logs
3. Perform threat intelligence extraction using ML from threat reports

UNIT-IV

Advanced AI Models and NLP in Healthcare Security: Generative Adversarial Networks (GANs), Variational Autoencoders (VAEs), synthetic data generation, anonymization techniques, privacy-preserving AI, differential privacy, deepfake detection, model inversion attacks, supervised learning for fraud detection, insider threat identification using NLP, AI-powered communication monitoring tools

Practice:

1. Build a generative model (GAN) for synthetic medical data
2. Configure access control and network security for a simulated hospital network
3. Evaluate the impact of adversarial attacks on healthcare AI systems

UNIT – V

Cybersecurity Operations and Compliance in Healthcare: Intrusion detection and prevention systems (IDPS), Endpoint Detection and Response (EDR), RBAC/ABAC, multi-factor authentication (MFA), SIEM tools, adversarial ML threats, model robustness, fairness and bias mitigation, forensic analysis, GDPR principles, AI-driven compliance monitoring, automated auditing protocols, post-incident review and modern breach prevention strategies

Practice:

1. Analyze a real-world healthcare breach and propose AI solutions
2. Build a dashboard for compliance monitoring using AI tools

Text Books:

1 Pascal Hitzler et al., *Foundations of AI in Healthcare*, Springer
2 Sumeet Dua and U. Rajendra Acharya, *Machine Learning in Healthcare Informatics*, Springer
3 HIPAA Journal – https://www.hipaajournal.com/

Reference Books:

1 Arvind Narayanan et al., *Security and Privacy in Machine Learning*, Princeton University Resources
2 Research papers from IEEE, Springer, Elsevier on AI in healthcare cybersecurity

3 OWASP Healthcare InfoSec resources – https://owasp.org/

Web Links:

1 https://www.microsoft.com/en-us/research/project/ai-for-health/

Advanced Topics in Generative AI

	L	T	P	C
Course Code:2501AI45	**2**	**0**	**1**	**3**

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

CO1: Explain the architectures and working of advanced GenAI models
CO2: Implement and fine-tune generative models for practical applications
CO3: Analyze and create multimodal generative systems
CO4: Evaluate ethical, legal, and societal implications of GenAI technologies
CO5: Design solutions using prompt engineering and domain-specific generative pipelines.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	3	2	2	2	2	1	2	2	-	-	2

CO2	3	3	3	3	3	2	1	-	-	-	3
CO3	3	3	3	3	3	1	2	-	1	-	2
CO4	2	2	2	2	1	3	3	3	1	-	2
CO5	3	3	3	3	3	1	2	2	1	-	3

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1	-	2
CO2	-	1
CO3	-	2
CO4	-	2
CO5	-	2

UNIT-I

Architectures in Generative AI: Generative Adversarial Networks (GANs), Variational Autoencoders (VAEs), Autoregressive models (PixelCNN, WaveNet), Denoising Diffusion Probabilistic Models (DDPM), Latent diffusion models (LDMs), Transformer-based text generation, GPT architecture and applications, T5 for text-to-text tasks, BERT for masked language modeling and generation, Ethical concerns in generative AI, Misuse of synthetic content, Hallucination and bias in AI models, Content authenticity and watermarking, Open-source vs closed-source model policies, Responsible AI development frameworks

Practice:

1. See differences between a VAE and a small DCGAN on MNIST
2. Generate text with a small GPT model and check for hallucinations / bias.

UNIT-II

Training & Fine-tuning GenAI Models: Pretraining vs fine-tuning paradigms, Transfer learning in NLP, Prompt tuning techniques, Prompt injection attacks, Prompt engineering best practices, Reinforcement Learning with Human Feedback (RLHF), Human-in-the-loop training, Model alignment and safety, Low-Rank Adaptation (LoRA), Parameter-Efficient Fine-Tuning (PEFT), Model robustness and adversarial prompts, Ethical considerations in LLM training, Bias and fairness in fine-tuning, Secure deployment of tuned models, Governance policies for generative AI

Practice:

1. See the difference between full fine-tuning and lightweight prompt-based tuning on a small NLP task
2. Understand how prompt injection can make an LLM ignore instructions and learn basic mitigation

UNIT-III

Multimodal Generative AI: Text-to-Image generation (DALL·E, Stable Diffusion), Text-to-Audio synthesis (e.g., Bark, Tortoise TTS), Text-to-Video models (e.g., Sora, Gen-2),

Diffusion models in multimodal generation, Latent space conditioning, Image captioning algorithms, Visual Question Answering (VQA) systems, Multimodal Transformers (e.g., Flamingo, GPT-4V), Cross-modal embedding techniques, Ethical concerns in synthetic media, Deepfake detection in multimodal content, Dataset biases in multimodal training, Content moderation policies, Accessibility through multimodal AI, Responsible deployment of generative models

Practice:

1. Generate images from text prompts using a pretrained diffusion model (Stable Diffusion mini / DALL·E API) and then generate captions for them using a pretrained image-captioning model.
2. Ask a pretrained Visual Question Answering (VQA) model questions about an image and check for biases or incorrect answers.

UNIT-IV

Applications and Industry Use Cases: AI-assisted content creation tools (e.g., Canva, Adobe Firefly), Generative design in digital media, AI in code generation (e.g., GitHub Copilot, CodeWhisperer), Game development with generative AI, Personalized learning with AI tutors, AI in medical imaging and diagnostics, Drug discovery using generative models, AI for scientific hypothesis generation, Responsible AI in education and healthcare, Data privacy in GenAI applications, Intellectual property and content ownership, Bias and fairness in AI-generated content, Human-AI

Practice:

1. Use a free AI-assisted tool (like Canva Text-to-Image or Adobe Firefly free plan) to create designs from text prompts and reflect on ownership, bias, and real-world use.
2. Use an AI code assistant (GitHub Copilot, CodeWhisperer, or Hugging Face CodeGen) to generate solutions for small problems, test outputs, and discuss safety/privacy.

UNIT – V

Ethical Considerations and Governance: Deepfakes and synthetic media threats, Disinformation and misinformation campaigns, AI alignment challenges, Bias in training data and model outputs, Hallucination in generative AI models, Content authenticity verification, Model interpretability and explainability, Copyright issues in AI-generated content, Legal frameworks for AI governance, International AI regulations (EU AI Act, U.S. EO), Ethics in AI development and deployment, Responsible AI and GenAI frameworks (e.g., OECD, NIST), Safety protocols for model deployment, Transparency and accountability in GenAI, Human oversight and control in AI systems

Practice:

1. Explore how deepfakes/synthetic media can be created and detected, and discuss the risks and governance measures.
2. Understand bias in model outputs and the importance of explainability & governance.

Text Books:

1 Goodfellow et al., Deep Learning, MIT Press, 2016

2 L. Hugging Face Team, The Transformers Book, Hugging Face, 2023
3 A. Radford et al., Language Models are Few-Shot Learners, OpenAI, 2020
4 P. Isola et al., Image-to-Image Translation with Conditional Adversarial Networks, CVPR, 2017

Reference Books:

1 C. Raffel et al., Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer, JMLR, 2020
2 OpenAI, GPT-4 Technical Report, arXiv, 2023
3 A. Dosovitskiy et al., An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale, ICLR, 2021
4 C. O'Neil, Weapons of Math Destruction, Crown Publishing, 2016

Web Links:

1 https://www.coursera.org/courses?query=generative%20ai
2 https://learn.microsoft.com/en-us/training/modules/fundamentals-generative-ai/

Ethical AI and Responsible Computing

Course Code:2501AI46	L	T	P	C
	2	0	1	3

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

CO1: Explain the ethical foundations and societal impact of AI and computing
CO2: Identify algorithmic bias, discrimination, and fairness concerns in AI systems
CO3: Apply ethical frameworks and principles to assess AI design and deployment
CO4: Design responsible AI solutions adhering to transparency and accountability
CO5: Analyze legal, regulatory, and real-world challenges in AI ethics.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	2	3	2	-	1	-	-	-	-	2	2
CO2	2	2	2	-	1	-	-	-	-	-	1
CO3	2	3	2	-	1	-	-	-	-	2	2
CO4	3	3	1	-	-	-	-	-	-	-	2
CO5	3	3	1	-	-	-	-	-	-	1	1

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO1	PSO2
CO1	-	2
CO2	-	2
CO3	-	2
CO4	-	2
CO5	-	2

UNIT-I

Introduction to Ethical AI: Algorithmic bias,AI transparency and explainability, Ethical hacking and penetration testing, Data privacy and consent, Deepfake and misinformation threats, Responsible AI principles, Digital surveillance ethics, Ethical governance frameworks, AI policy and regulatory compliance

Practice:
1. Analyze ethical dilemmas in emerging AI use cases

UNIT-II

Bias and Fairness in AI: Algorithmic bias detection, Fairness metrics evaluation, Discrimination in automated decisions, Bias mitigation techniques, Ethical AI principles, Fairness-aware machine learning tools, Regulatory and policy frameworks, Inclusive data practices, Real-world case studies of algorithmic bias

Practice:

1. Identify bias in datasets and explore fairness metrics

UNIT-III

Ethical Frameworks and Principles: Ethical decision-making frameworks, AI moral reasoning models, Deontological rule-based systems, Utilitarian outcome optimization, Virtue ethics in AI behavior, Human-centered design principles, Value alignment in AI systems, Ethical risk assessment protocols, Policy guidelines for ethical AI

Practice:

1. Evaluate AI algorithms using ethical impact assessment tools
2. Conduct an ethical risk analysis on a public AI system

UNIT-IV

Responsible AI Design: Explainability techniques in AI, Model interpretability tools, Accountability in automated decisions, Transparency principles in AI systems, Auditing protocols for AI models, Ethical threats of black-box algorithms, Regulatory policies for XAI, Human-in-the-loop frameworks, Trust and reliability in AI outputs

Practice:

1. Design an interpretable AI model and generate explanations using XAI
2. Develop a code of ethics or policy guideline for responsible AI use

UNIT – V

Governance, Policy, and Regulation: EU AI Act provisions, Global AI regulatory frameworks, Data privacy and protection laws, Compliance auditing tools, Risk-based AI classification, Legal and ethical AI principles, Enforcement protocols for AI governance, Cross-border data sharing threats, Challenges in regulatory compliance

Ethical Challenges in Real-world Applications: Biometric data protection, AI surveillance ethics, Deepfake detection technologies, Autonomous decision-making protocols, Privacy threats in facial recognition, Regulatory frameworks for AI surveillance, AI misuse and societal impact, Security vulnerabilities in AV systems, Legal and ethical governance of deepfakes

Practice:

1. Simulate a case study review under AI regulation (e.g., GDPR or EU AI Act)
2. Explore adversarial impacts of unregulated AI systems (e.g., misinformation, surveillance)
3. Debate a controversial AI deployment and propose ethical alternatives

TextBooks:

1 Virginia Dignum, *Responsible Artificial Intelligence: How to Develop and Use AI in a Responsible Way*, Springer
2 Markus D. Dubber, Frank Pasquale, Sunit Das, *Oxford Handbook of Ethics of AI*, Oxford University Press
3 Shannon Vallor, *Technology and the Virtues: A Philosophical Guide to a Future Worth Wanting*, Oxford University Press

Reference Books:

1 IEEE Global Initiative on Ethics of Autonomous and Intelligent Systems
2 World Economic Forum – AI Governance Frameworks
3 AI Now Institute Reports – https://ainowinstitute.org/

Web Links:

1 https://www.microsoft.com/en-us/ai/responsible-ai#:~:text=Microsoft's%20responsible%20AI%20principles%20include,inclusiveness%2C%20transparency%2C%20and%20accountability.