PROGRAM CURRICULUM

(Applicable for the batches admitted from A.Y 2026-27)

THREE YEAR U.G PROGRAM

BACHELOR OF COMPUTER APPLICATIONS

ADITYA

UNIVERSITY

Aditya Nagar, ADB Road, Surampalem - 533 437

VISION & MISSION OF THE UNIVERSITY

VISION:
To be a globally recognized University through excellence in Education, Innovation and Sustainable growth.

MISSION:
Deliver collaborative education to prepare students for global challenges through

- Transformative learning
- Vibrant research ecosystem
- Sustainable and inclusive community

VISION & MISSION OF THE DEPARTMENT

VISION:
To become a distinguished center of excellence in computer applications, driving innovation, education, and technological advancement for societal benefit.

MISSION:

M1: Deliver transformative and industry-relevant education through cutting-edge pedagogy and collaborative learning.

M2: Advance impactful research in computing, fostering innovation and knowledge creation.

M3: Develop sustainable and inclusive technological solutions that address societal and global challenges.

PROGRAM OUTCOMES (POs)

PO1: **Computational Knowledge**

Apply knowledge of mathematics, computing fundamentals, and domain-specific concepts to analyze and solve problems in computer applications.

PO2: **Problem Analysis**

Identify, analyze, and formulate solutions to complex computing problems using principles of mathematics, computer science, and application-oriented knowledge.

PO3: **Design and Development of Solutions**

Design, develop, test, and evaluate software solutions and computer-based systems that meet specified requirements, considering societal, environmental, health, safety, and legal aspects.

PO4: **Modern Tool Usage**

Create, select, and apply appropriate techniques, resources, and modern IT tools (including programming frameworks, databases, cloud platforms, and AI tools) for developing computing solutions, with an understanding of their limitations.

PO5: **Individual and Teamwork**

Function effectively as an individual, team member, or team leader in diverse and multidisciplinary environments to accomplish common goals.

PO6: **Communication**

Communicate effectively with computing professionals and society by producing clear technical documentation, reports, and presentations.

PO7: **Ethics**

Apply ethical principles and commit to professional ethics, responsibilities, and norms of computing practices.

PO8: **Society and Sustainability**

Understand the impact of computer applications and IT solutions in societal, environmental, and sustainability contexts, and contribute to sustainable development.

PO9: **Project Management**

Apply computing knowledge and management principles to plan, execute, and manage projects effectively in multidisciplinary environments.

PO10: **Lifelong Learning**

Recognize the need for, and engage in, independent and lifelong learning to adapt to emerging technologies and professional practices.

PROGRAM SPECIFIC OUTCOMES (PSOS)

PSO1: Software Development and Computing Applications

Apply programming principles, data structures, databases, and modern development frameworks to design, implement, test, and deploy reliable software applications

PSO2: Emerging Technologies and IT Solutions

Utilize contemporary technologies such as cloud computing, data analytics, web/mobile platforms, and basic AI tools to develop scalable, secure, and sustainable IT solutions.

PROGRAM EDUCATIONAL OBJECTIVES (PEOS)

PEO1: Professional Competence

Graduates will establish themselves as competent computing professionals by applying core computer application knowledge and modern tools to develop effective software solutions for industry and societal needs.

PEO2: Higher Education and Lifelong Learning

Graduates will pursue higher education, professional certifications, research, or continuous skill development to adapt to emerging technologies and global computing challenges.

PEO3: Ethics, Sustainability, and Societal Contribution

Graduates will demonstrate ethical responsibility, professional integrity, and social awareness while contributing to sustainable and inclusive technological development.

SCHOOL OF ENGINEERING

Department of Computer Applications

Program Curriculum – 2026

(**Applicable for the batches admitted from A. Y. 2026-27**)

U.G. Programs Offered

- Bachelor of Computer Applications

P.G. Programs Offered

- Master of Computer Applications

ADITYA UNIVERSITY

DEPARTMENT OF COMPUTER APPLICATIONS

Bachelor of Computer Applications

(BCA) Program Curriculum - 2026

Credit Division:

Sl.No.	Category of the Course	Credits
1	Program Core Courses (PCC)	70
2	Discipline Specific Elective Courses (DSE)	24
3	Ability Enhancement Courses (AEC)	04
4	Multi - Disciplinary Courses (MDC)	02
5	Value Added Courses (VAC)	08
6	Mandatory Courses (MC)	01
7	Summer Internship (SI)	03
8	Project (PROJ)	08
	Total Credits	**120**

Program Core Courses (PCC)

Course Code	Course Title	Level	L	T	P	C	CIE	SEE	Total	Pre-Requisite
2603MC01	Fundamentals of Mathematics - I	FC	2	1	0	3	50	50	100	-
2603MC02	Digital Computer Fundamentals	FC	1	0	2	3	50	50	100	-
2603MC03	Problem Solving Techniques using C	FC	3	0	2	5	50	50	100	-
2603MC04	Database Management Systems	FC	3	0	2	5	50	50	100	-
2603MC05	Fundamentals of Mathematics - II	FC	2	1	0	3	50	50	100	FM-I
2603MC06	Object Oriented Programming using C++	FC	3	0	2	5	50	50	100	C
2603MC07	Software Engineering	FC	3	0	0	3	50	50	100	-
2603MC08	Data Structures	FC	2	0	2	4	50	50	100	C
2603MC09	Operating Systems	FC	2	0	1	3	50	50	100	-
2603MC10	Probability and Statistics	IC	2	1	0	3	50	50	100	-
2603MC11	Web Technologies	IC	1	0	2	3	50	50	100	-
2603MC12	Computer Architecture	IC	3	0	0	3	50	50	100	-
2603MC13	Java Programming	IC	2	0	2	4	50	50	100	-
2603MC14	Artificial Intelligence	IC	2	0	2	4	50	50	100	-
2603MC15	Python Programming	IC	3	0	2	5	50	50	100	-
2603MC16	Design and Analysis of Algorithms	IC	3	1	0	4	50	50	100	-
2603MC17	Computer Networks	AC	3	0	2	5	50	50	100	-
2603MC18	Machine Learning with Python	AC	3	0	2	5	50	50	100	Python
Total Credits			**43**	**4**	**23**	**70**				

Discipline Specific Elective Courses (DSE)

Software Development & Security

Course Code	Course Title	Level	L	T	P	C	CIE	SEE	Total	Pre-Requisite
2603MC19	Object Oriented Software Engineering	IC	3	0	1	4	50	50	100	SE
2603MC20	Software Project Management	IC	3	0	1	4	50	50	100	SE
2603MC21	Agile Methodologies	AC	2	0	2	4	50	50	100	SE
2603MC22	Software Testing Management	AC	3	0	1	4	50	50	100	-
2603MC23	Ethical Hacking	AC	3	0	1	4	50	50	100	-
2603MC24	Block Chain Technologies	AC	3	0	1	4	50	50	100	-

Data Science & AI

Course Code	Course Title	Level	L	T	P	C	CIE	SEE	Total	Pre-Requisite
2603MC25	Fundamentals of Data Science	IC	2	0	2	4	50	50	100	Python
2603MC26	Data Analytics	IC	3	0	1	4	50	50	100	-
2603MC27	Expert Systems	AC	3	0	1	4	50	50	100	-
2603MC28	Deep Learning	AC	3	0	1	4	50	50	100	MLP
2603MC29	Information Retrieval Systems	AC	3	1	0	4	50	50	100	-
2603MC30	Foundations of Data Engineering	AC	3	0	1	4	50	50	100	-

Cloud & IoT Applications

Course Code	Course Title	Level	L	T	P	C	CIE	SEE	Total	Pre-Requisite
2603MC31	Cloud Computing	IC	3	0	1	4	50	50	100	-
2603MC32	Mobile Application Development	IC	2	0	2	4	50	50	100	-
2603MC33	NoSQL Databases	AC	2	0	2	4	50	50	100	-
2603MC34	Internet of Things	AC	2	0	2	4	50	50	100	-
2603MC35	Data Visualization	AC	3	0	1	4	50	50	100	-
2603MC36	Business Intelligence & Analytics	AC	3	0	1	4	50	50	100	-

Ability Enhancement Courses (AEC)

Course Code	Course Title	Level	L	T	P	C	CIE	SEE	Total	Pre-Requisite
2603MC37	Fundamental English Proficiency	FC	0	0	1	1	100	-	100	-
2603MC38	Advanced English Proficiency	IC	0	0	1	1	100	-	100	FEP
2603MC39	Student Activity Based Learning	AC	0	0	2	2	100	-	100	-

Multi - Disciplinary Courses (MDC)

Course Code	Course Title	Level	L	T	P	C	CIE	SEE	Total	Pre-Requisite
2603MC40	Design Thinking & Innovation	IC	0	0	2	2	100	-	100	-

Value Added Courses (VAC)

Course Code	Course Title	Level	L	T	P	C	CIE	SEE	Total	Pre-Requisite
2603MC41	Universal Human Values	FC	2	0	0	2	100	-	100	-
2603MC42	Environmental Science & Sustainability	FC	2	0	0	2	100	-	100	-
2603MC43	Indian Knowledge Systems	IC	2	0	0	2	100	-	100	-
2603MC44	Indian Constitution	IC	2	0	0	2	100	-	100	-

Mandatory Courses (MC)

Course Code	Course Title	Level	L	T	P	C	CIE	SEE	Total	Pre-Requisite
2603MC45	Employability Skills - I	FC	0	0	1	0	100	-	100	-
2603MC46	Employability Skills - II	FC	0	0	1	0	100	-	100	ES - I
2603MC47	Employability Skills - III	IC	0	0	1	0	100	-	100	ES - II
2603MC48	Employability Skills - IV	IC	0	0	1	1	100	-	100	ES - III

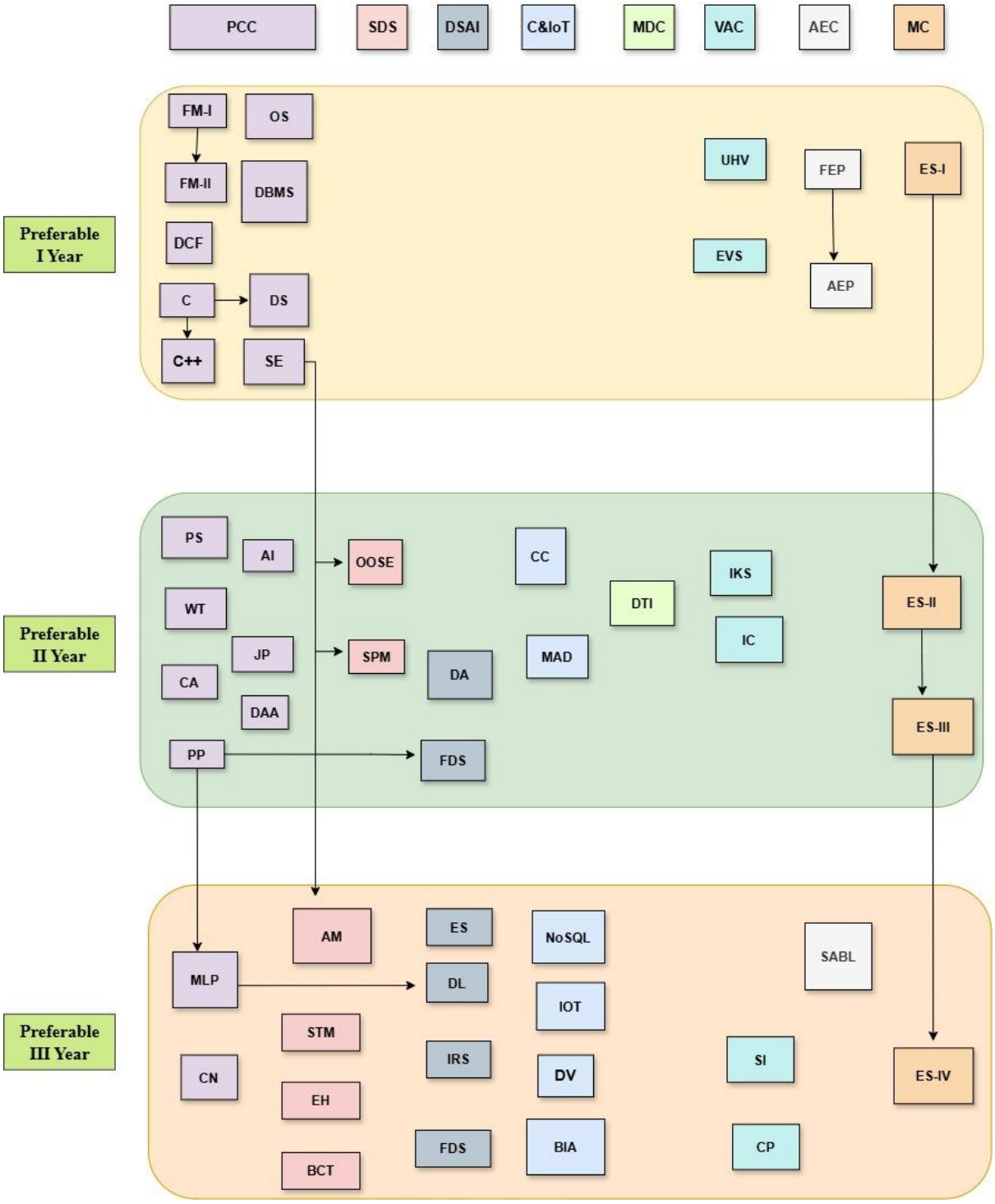
Summer Internship (SI)

Course Code	Course Title	Level	L	T	P	C	CIE	SEE	Total	Pre-Requisite
2603MC49	Summer Internship	IC	0	0	3	3	100	-	100	-

Project (PROJ)

Course Code	Course Title	Level	L	T	P	C	CIE	SEE	Total	Pre-Requisite
2603MC50	Capstone Project	AC	0	0	8	8	50	50	100	-

2026 BCA CURRICULAM
PREREQUISITE FLOW CHART



Preferable I Year Courses	
FM-I	Fundamentals of Mathematics – I
DCF	Digital Computer Fundamentals
C	Problem Solving Techniques using C
DBMS	Database Management Systems
FEP	Fundamental English Proficiency
UHV	Universal Human Values
FM-II	Fundamentals of Mathematics - II
C++	Object Oriented Programming using C++
SE	Software Engineering
DS	Data Structures
OS	Operating Systems
AEP	Advanced English Proficiency
EVS	Environmental Science & Sustainability
ES-I	Employability Skills – I

Preferable II Year Courses	
PS	Probability and Statistics
WT	Web Technologies
CA	Computer Architecture
JP	Java Programming
IKS	Indian Knowledge Systems
ES-II	Employability Skills – II
AI	Artificial Intelligence
PP	Python Programming
DAA	Design and Analysis of Algorithms
IC	Indian Constitution
DTI	Design Thinking & Innovation
ES-III	Employability Skills – III
OOSE	Object Oriented Software Engineering
SPM	Software Project Management
FDS	Fundamentals of Data Science
DA	Data Analytics
CC	Cloud Computing
MAD	Mobile Application Development

Preferable III Year Courses	
CN	Computer Networks
MLP	Machine Learning with Python
ES-IV	Employability Skills - IV
SI	Summer Internship
SABL	Student Activity Based Learning
CP	Capstone Project
AM	Agile Methodologies
STM	Software Testing Management
EH	Ethical Hacking
BCT	Block Chain Technologies
ES	Expert Systems
DL	Deep Learning
IRS	Information Retrieval Systems
FDE	Foundations of Data Engineering
NoSQL	NoSQL Databases
IoT	Internet of Things
DV	Data Visualization
BIA	Business Intelligence & Analytics

Suggestive Semester wise Curriculum
I Semester

Course Code	Course Title	Level	Course Category	Credits				Total Hours
				L	T	P	Total	
2603MC01	Fundamentals of Mathematics - I	FC	PCC	2	1	0	3	3
2603MC02	Digital Computer Fundamentals	FC	PCC	1	0	2	3	5
2603MC03	Problem Solving Techniques using C	FC	PCC	3	0	2	5	7
2603MC04	Database Management Systems	FC	PCC	3	0	2	5	7
2603MC37	Fundamental English Proficiency	FC	AEC	0	0	1	1	2
2603MC41	Universal Human Values	FC	VAC	2	0	0	2	2
			Total	**11**	**1**	**7**	**19**	**26**

II Semester

Course Code	Course Title	Level	Course Category	Credits				Total Hours
				L	T	P	Total	
2603MC05	Fundamentals of Mathematics - II	FC	PCC	2	1	0	3	3
2603MC06	Object Oriented Programming using C++	FC	PCC	3	0	2	5	7
2603MC07	Software Engineering	FC	PCC	3	0	0	3	3
2603MC08	Data Structures	FC	PCC	2	0	2	4	6
2603MC09	Operating Systems	FC	PCC	2	0	1	3	4
2603MC38	Advanced English Proficiency	IC	AEC	0	0	1	1	2
2603MC42	Environmental Science & Sustainability	FC	VAC	2	0	0	2	2
2603MC45	Employability Skills - I	FC	MC	0	0	1	0	2
			Total	**14**	**1**	**7**	**21**	**29**

III Semester

Course Code	Course Title	Level	Course Category	Credits				Total Hours
				L	T	P	Total	
2603MC10	Probability and Statistics	IC	PCC	2	1	0	3	3
2603MC11	Web Technologies	IC	PCC	1	0	2	3	5
2603MC12	Computer Architecture	IC	PCC	3	0	0	3	3
2603MC13	Java Programming	IC	PCC	2	0	2	4	6
	Professional Elective - I	IC	DSE				4	5
2603MC43	Indian Knowledge Systems	IC	VAC	2	0	0	2	2
2603MC46	Employability Skills - II	FC	MC	0	0	1	0	2
			Total	**10**	**1**	**5**	**19**	**26**

IV Semester

Course Code	Course Title	Level	Course Category	Credits				Total Hours
				L	T	P	Total	
2603MC14	Artificial Intelligence	IC	PCC	2	0	2	4	6
2603MC15	Python Programming	IC	PCC	3	0	2	5	7
2603MC16	Design and Analysis of Algorithms	IC	PCC	3	1	0	4	4
	Professional Elective - II	IC	DSE				4	5
2603MC44	Indian Constitution	IC	VAC	2	0	0	2	4
2603MC40	Design Thinking & Innovation	IC	MDC	0	0	2	2	4
2603MC47	Employability Skills - III	IC	MC	0	0	1	0	2
			Total	**10**	**1**	**7**	**21**	**32**

V Semester

Course Code	Course Title	Level	Course Category	Credits				Total Hours
				L	T	P	Total	
2603MC17	Computer Networks	AC	PCC	3	0	2	5	7
2603MC18	Machine Learning with Python	AC	PCC	3	0	2	5	7
	Professional Elective - III	AC	DSE				4	4
	Professional Elective - IV	AC	DSE				4	4
2603MC48	Employability Skills - IV	IC	MC	0	0	1	1	2
2603MC49	Summer Internship	IC	SI	0	0	3	3	0
			Total	**6**	**0**	**8**	**22**	**24**

VI Semester

Course Code	Course Title	Level	Course Category	Credits				Total Hours
				L	T	P	Total	
	Professional Elective – V	AC	DSE				4	4
	Professional Elective – VI	AC	DSE				4	4
2603MC39	Student Activity Based Learning	AC	AEC	0	0	2	2	0
2603MC50	Capstone Project	AC	PROJ	0	0	8	8	0
			Total	**0**	**0**	**10**	**18**	**8**

FUNDAMENTALS OF MATHEMATICS - I

Course Code: 2603MC01

L	T	P	C
2	1	0	3

Course Outcomes:

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

CO1: Apply concepts of elementary operations to find the rank of a matrix

CO2: Apply the Cayley Hamilton theorem to find the inverse and higher powers of a matrix

CO3: Apply various interpolation techniques to estimate unknown value between discrete data points

CO4: Apply numerical integration techniques to approximate definite integrals

CO5: Find solution of equations using numerical methods.

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	PO9	PO10
CO1	3	2	-	-	-	-	-	-	-	-
CO2	3	2	-	-	-	-	-	-	-	-
CO3	3	2	-	-	-	-	-	-	-	-
CO4	3	2	-	-	-	-	-	-	-	-
CO5	3	2	-	-	-	-	-	-	-	-

UNIT – I

Matrices: Determinants, properties of determinants, matrices, matrices operations, transpose of a matrix, adjoint of a square matrix, inverse of a matrix, rank of a matrix - Eechelon form, Normal form.

UNIT – II

Solution of linear system of equations: Cramer's rule, matrix inversion method, Test the Consistency of linear system of equations, Eigen values and Eigen vectors, Cayley-Hamilton theorem (without proof), Inverse and powers of a square matrix.

UNIT – III

Interpolation: Operators, Forward and Backward Difference Operations and Their Interrelation. Interpolation Formulae: Newton's Forward, Backward and Lagrange's Formula

UNIT – IV

Numerical differentiation &Integration: Numerical Differentiation: Formulae for derivatives of first and second order. **Numerical Integration**: Trapezoidal rule, Simpson's 1/3 rule, Simpson's 3/8 rule.

UNIT – V

Solution of Equations by using numerical methods: Bisection Method, False Position Method, Newton Raphson method, Jacobi's iteration method, Gauss-Siedel iteration method.

Textbooks:

1. B.S. Grewal, Higher Engineering Mathematics, 45th Edition, Khanna Publishers, ISBN: 9788194742579
2. S.S. Sastry, Introductory Methods of Numerical Analysis, 5th Edition, PHI Learning Pvt. Ltd., ISBN: 9788120345928

Reference Books:

1. M.K. Jain, S.R.K. Iyengar & R.K. Jain, Numerical Methods for Scientific and Engineering Computation, 6th Edition, New Age International Publishers, ISBN: 9788122430967
2. B.V. Ramana, Higher Engineering Mathematics, 1st Edition, Tata McGraw Hill Education, ISBN: 9781259064401

Web Links:

1. https://www.nptelvideos.com/video.php?id=111
2. https://ocw.mit.edu/courses/18-06-linear-algebra-spring-2010/resources/lecture-21-eigenvalues-and-eigenvectors/
3. https://freevideolectures.com/course/4617/nptel-numerical-methods/20
4. https://www.youtube.com/watch?v=0rtaUUonwkU

DIGITAL COMPUTER FUNDAMENTALS

Course Code: 2603MC02

L	T	P	C
1	0	2	3

Course Outcomes:

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

CO1: Describe the concepts and importance of office automation tools in modern organizations.

CO2: Create and format professional documents using word processing tools.

CO3: Apply spreadsheet tools for data organization, calculation, and analysis.

CO4: Design effective presentations using presentation software.

CO5: Utilize office automation tools for routine office operations and communication.

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	PO9	PO10
CO1	2	1	-	-	-	-	-	-	-	-
CO2	2	2	-	2	-	1	-	-	-	-
CO3	2	2	-	2	-	-	-	-	-	-
CO4	1	-	-	2	1	2	-	-	-	-
CO5	2	1	-	2	1	1	-	-	-	2

Mapping of Course Outcomes with Program Specific Outcomes:

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

UNIT – I

Fundamentals of Digital Office Applications: Overview of computer hardware and system software, Office Applications: meaning, objectives and advantages, Components of office Application systems, Introduction to operating systems with special reference to Windows, File and folder management, Introduction to office automation software packages.

Practice:

1. Create folders and subfolders and perform basic file operations.
2. Use Windows operating system features to open and manage applications.
3. Perform file operations such as copy, move, rename, and delete files.

UNIT – II

Word Processing Tools: Introduction to word processing software (MS Word), Creating, editing, saving and opening documents. Text formatting, paragraph formatting, page setup. Headers and footers, page numbering. Creating and formatting tables, MS Word – Images and Graphics. find and replace, Mail merge.

Practice:

1. Create an official letter using proper text and paragraph formatting.
2. Prepare a report with page setup, header, footer, and page numbering.
3. Design a resume using tables and formatting tools.
4. Insert images and graphics into a document and format them appropriately.
5. Perform a mail merge operation to generate multiple documents.

UNIT – III

Spreadsheet Tools and Data Analysis: Introduction to spreadsheets, Overview of Ribbon Layout, Workbook and worksheet concepts, File Saving and Sharing, Data entry and editing, Cell referencing, Formulas and functions: SUM, AVERAGE, COUNT, MAX, MIN, IF. Creating and formatting charts. Sorting and filtering data.

Practice:

1. Create a student marksheet using basic formulas.
2. Apply SUM, AVERAGE, COUNT, MAX, MIN, and IF functions on given data.
3. Generate charts for the given dataset.
4. Perform sorting operations on spreadsheet data.
5. Apply filtering techniques to display required records.

UNIT – IV

Advanced Spreadsheet Applications: Advanced spreadsheet features: Relative, absolute and mixed cell referencing. Advanced functions and formulas: INDEX, MATCH, SUMIFS, COUNTIFS. Importing Data from various sources, Dynamic Range and Named Ranges, Chart customization. Printing worksheets and charts.

Practice:

1. Prepare a salary statement with allowances and deductions.
2. Apply relative, absolute, and mixed cell referencing in formulas.
3. Use advanced functions such as INDEX, MATCH, SUMIFS, and COUNTIFS.
4. Import data from external sources into a spreadsheet.
5. Create and customize charts for the given data.

UNIT – V

Presentation Tools: Introduction to presentation software. Creating presentations using templates and themes. Slide layouts and master slides. Slide duplicate. Inserting text, images, tables and charts. Animations and slide transitions. Running slide shows and printing presentations.

Practice:

1. Create a power point presentation for a seminar or meeting.
2. Apply suitable slide layouts and master slides in power point.
3. Insert text, images, tables, and charts into slides.
4. Apply animations and slide transitions effectively.
5. Design an invitation for a seminar or college event using presentation software.

Additional Practice:

1. Complete a mini project based on common office automation tasks.
2. Prepare an office report by creating documents, analyzing data, and presenting results using office tools.
3. Perform data management tasks using word processing, spreadsheet, and presentation software together.

Textbooks:

1. Pradeep K. Sinha & Priti Sinha, Computer Fundamentals, 6th Edition, BPB Publications, ISBN: 978-9353160841
2. Reema Thareja, Fundamentals of Computers, 1st Edition, Oxford University Press, ISBN: 978-0199456789

Reference Books:

1. V. Rajaraman, Fundamentals of Computers, 5th Edition, PHI Learning Private Limited, ISBN: 978-8120345752
2. Peter Norton, Introduction to Computers, 7th Edition, McGraw Hill Education, ISBN: 978-0070634496

Web Links:

1. https://www.geeksforgeeks.org/computer-science-fundamentals/what-are-office-tools
2. https://www.w3schools.com/excel/index.php
3. https://learn.microsoft.com/en-us/sharepoint/dev/general-development/powerpoint-automation-services-in-sharepoint

PROBLEM SOLVING TECHNIQUES USING C

Course Code: 2603MC03

L	T	P	C
3	0	2	5

Course Outcomes:

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

CO1: Understand & apply algorithms, flowcharts & structured programming concepts in C.

CO2: Implement decision control structures and loops for program logic development.

CO3: Use arrays and strings efficiently for data storage and manipulation.

CO4: Develop modular C programs using functions, recursion, arrays as parameters, pointers, and variable scope concepts.

CO5: Design C programs using structures and unions and performing file operations for data storage and retrieval.

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	PO9	PO10
CO1	3	2	1	-	-	-	-	-	-	-
CO2	1	3	2	-	-	-	-	-	-	-
CO3	2	2	3	1	-	-	-	-	-	-
CO4	2	2	3	1	-	-	-	-	-	-
CO5	2	2	2	3	-	-	-	-	1	-

Mapping of Course Outcomes with Program Specific Outcomes:

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

UNIT – I

Introduction to Algorithms and Programming Languages: Algorithm – Key Features of Algorithms, Examples of Algorithms, Flowcharts, Pseudocode, Programming Languages, Generation of Programming Languages, Structured Programming Language.

Introduction to C: Structure of a C Program, Writing the First C Program, Files in C, Compiling and Executing C programs, Comments, Keywords, Identifiers, Basic Data Types, Variables, Constants, I/O Statements, Operators, Type Conversion and Type Casting.

Practice:

1. Write a C program to print "Hello World".
2. Write a program to add two integers and display the result.
3. Write a C program to calculate the area of a circle.
4. Write a C program to demonstrate type conversion (implicit and explicit).

UNIT – II

Decision Control Statements: Simple if, if-else, else-if ladder, nested if else, Multi way branching statements. **Iterative Statements**: while, do-while, for loops, Nested Loops. **Jumping Statements:** Break and Continue Statements, Goto Statement.

Practice:

1. Write a C program to find whether a number is positive, negative or zero using if-else.
2. Write a C program to print the weekdays by using switch case.
3. Write a C program to find the factorial of a number using a while loop.
4. Write a C program to print multiplication table of a number using do-while loop.
5. Write a C program to print pyramids of stars by using for loop.
6. Write a C program to demonstrate the use of break and continue statements.

UNIT – III

Arrays: Introduction, Declaration of Arrays, Accessing Elements, Storing Values, Calculating Length, Array Operations, One-Dimensional Arrays, Two-Dimensional Arrays.

Strings: Introduction, String Operations using String Functions.

Practice:

1. Write a C program to read and print elements of a 1D array.
2. Write a program to calculate the sum and average of elements in an array.
3. Write a program to perform matrix addition (2D array).
4. Write a program to perform matrix multiplication (2D array).
5. Write a program to find given string is palindrome or not.

UNIT – IV

Functions: Introduction to functions, Function Declaration and Definition, Function Call, Return Statement, Passing Parameters, Recursive Functions, Passing One-Dimensional Array to Functions, Scope of Variables, Storage Classes.

Pointers: Understanding Computer Memory, Introduction to Pointers, Declaring Pointer Variables, Dynamic Memory Allocation, Drawbacks of Pointers.

Practice:

1. Write a program for swapping of two numbers by using call by value and call by reference.
2. Write a program to pass an array to a function and find the maximum value.
3. Write a recursive function to find the Fibonacci series up to n terms.
4. Write a program to demonstrate local and global variables.

UNIT – V

Structures and Unions: Introduction to Structures, Nested Structures, Introduction to Unions, Accessing Union Elements.

File Handling: Introduction to Files, Using Files in C, Reading and Writing Data from Files.

Practice:

1. Write a program to define a structure for student details and display them.
2. Write a program using nested structures for employee and address.
3. Write a program to read from and write to a file (text file).

Additional Practice:

1. Write a C program to reverse a number and check whether it is a palindrome
2. Write a C program to sort an array in ascending order using pointers.
3. Write a C program to find the second largest element in an array.

Textbooks:

1. E. Balagurusamy, Programming in ANSI C, 8th Edition, McGraw Hill, ISBN: 978-9353165136
2. Brian W. Kernighan & Dennis M. Ritchie, The C Programming Language, 2nd Edition, Pearson, ISBN: 978-0131103627

Reference Books:

1. Yashavant Kanetkar, Let Us C, 16th Edition, BPB Publications, ISBN: 978-9387287179
2. R.S. Salaria, Problem Solving and Programming in C, 2nd Edition, Khanna Book Publishing, ISBN: 978-9386173459

Web Links:

1. https://www.learn-c.org
2. https://nptel.ac.in/courses/106/105/106105171

DATABASE MANAGEMENT SYSTEMS

Course Code: 2603MC04

L	T	P	C
3	0	2	5

Course Outcomes:

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

CO1: understand how modern database systems store, organize, and manage data using data models and SQL.

CO2: Students will learn the basics of database history and design simple databases using ER diagrams for real-world systems.

CO3: Students will learn to use SQL commands to create tables and retrieve data using basic queries, conditions, and joins.

CO4: Able to analyze and improve database schemas using normalization techniques

CO5: understand how transactions are managed and how data is stored and indexed in a database

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	PO9	PO10
CO1	3	2	2	-	1	1	1	-	-	2
CO2	2	2	-	-	-	-	2	2	-	2
CO3	-	3	2	-	1	-	-	-	1	-
CO4	3	-	2	-	-	1	1	1	-	-
CO5	-	2	-	3	1	1	-	-	-	2

Mapping of Course Outcomes with Program Specific Outcomes:

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

UNIT – I

Introduction: Data base System Applications, data base System VS file System, Advantages of a DBMS, View of Data, Data Abstraction, instances and Schemas, data Models, the ER Model, Relational Model, Other Models, Database Languages: DDL, DML, DCL. Different types of data base Users and their responsibility.

Practice:

1. Implementation of DDL, DML commands of SQL with suitable examples
2. Implementation of different types of operators in SQL

UNIT – II

History of Data base Systems: Data base design and ER diagrams, Beyond ER Design Entities, Attributes and Entity sets, Relationships and Relationship sets, Additional features of ER Model, Concept Design with the ER Model, and Conceptual Design for Large enterprises.

Practice:

1. Draw an ER diagram for a student–Course Registration System.
2. Draw an ER diagram for an Online Shopping System

UNIT – III

Database Query Operation: DML operations- Procedural (selection, projection, set difference, cartesian product, join) and non-procedural: domain and tuple calculus, DDL operations. Basic SQL querying (select and project) using where clause, sub queries, grouping, aggregation, ordering, implementation of different types of joins.

Practice:

1. Study and Implementation of Group By & having clause and order by clause
2. Implementation of different types of aggregation functions?

UNIT – IV

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

Practice:

1. A relation STUDENT_MARKS is given below:
   STUDENT_MARKS (Roll No, Student Name, CourseID, CourseName, FacultyName, Marks)
   - Find the candidate key.
   - Check whether the relation is in 1NF, 2NF, and 3NF.

UNIT – V

Transaction Management: Transaction, properties of transactions, transaction log, and transaction management with SQL using commit rollback and save point.

Storage and Indexing: Database file organization, file organization on disk, heap files and sorted files, hashing, single and multi-level indexes.

Practice:

1. Write SQL queries to create an EMPLOYEE table, insert records, update salary, and demonstrate the use of COMMIT and ROLLBACK to show permanent and temporary transaction effects.

Additional Practice:

1. To understand database backup and recovery concepts using SQL commands**.**

Textbooks:

1. Database Management Systems, Raghu Ramakrishnan & Johannes Gehrke,3rd Edition, McGraw-Hill Education, ISBN: 978-0072465631
2. Database Management Systems Concepts, Ramez Elmasri, Shamkant B. Navathe, 7th Edition, Pearson Education, ISBN: 978-0133970777

Reference Books:

1. Concepts of Database Management Systems, Abraham Silberschatz, Henry F. Korth, S. Sudarshan, 6th Edition, McGraw-Hill Education, ISBN: 978-0078022159
2. Database Systems: A Practical Approach to Design, Implementation, and Management, Thomas Connolly & Carolyn E. Begg,7th Edition, Pearson Education, ISBN: 978-1292061184.

Web Links:

1. https://www.db-book.com/?utm_source=chatgpt.com
2. https://link.springer.com/book/10.1007/978-3-540-48399

FUNDAMENTALS OF MATHEMATICS - II

Course Code: 2603MC05

L	T	P	C
2	1	0	3

Course Outcomes:

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

CO1: Compute transitive closure, equivalence classes of binary relations

CO2: Differentiate various functions and solve counting problems.

CO3: Apply the principles of mathematical logic to statement calculus and predicate calculus.

CO4: Differentiate various types of Graphs

CO5: 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	PO9	PO10
CO1	3	2	-	-	-	-	-	-	-	-
CO2	3	2	-	-	-	-	-	-	-	-
CO3	3	2	-	-	-	-	-	-	-	-
CO4	3	2	-	-	-	-	-	-	-	-
CO5	3	2	-	-	-	-	-	-	-	-

UNIT – I

Set Theory and Relations: Sets, Set Operations, Algebra of Sets, Classes of Sets, Power Sets, Partitions, Relations, n-ary Relations, Representations of Relations, Composition of Relations, Types of Relations, Equivalence relation and classes, Transitive Closure, Partial Ordering Relations, Hasse Diagrams.

UNIT – II

Functions and Counting: Functions, One-to-One, onto and Invertible Functions, Mathematical, Exponential, Logarithmic Functions and Recursive functions, Basic Counting Principles, Permutations, Combinations, The Pigeonhole Principle, The Inclusion –Exclusion Principle

UNIT – III

Logic and Propositional Calculus: Propositions and Truth Tables, Tautologies, Logical Equivalence, Algebra of Propositions, Arguments, Logical Implication, Propositional Functions, Quantifiers, Inference Theory of Predicate Calculus.

UNIT – IV

Graph Theory: Graph, Subgraphs, Graph Representations: Adjacency and Incidence Matrices, Isomorphic Graphs, Paths and Circuits, Eulerian and Hamiltonian Graph.

UNIT – V

Multigraphs: Multigraphs, Bipartite and Planar Graphs, Euler's Theorem, Graph Colouring, Chromatic Number, Spanning Trees, BFS and DFS Algorithms, Minimal spanning trees, Prim's and Kruskal's Algorithms.

Textbooks:

1. J. P. Tremblay & P. Manohar, Discrete Mathematical Structures with Applications to Computer Science, 1st Edition, Tata McGraw Hill, ISBN: 9780074622353
2. S. Santha & E. V. Prasad, Mathematical Foundations of Computer Science, 1st Edition, Cengage Learning, ISBN: 9788131518588

Reference Books:

1. S. K. Sarkar, Mathematical Foundations of Computer Science, 1st Edition, S. Chand Publications, ISBN: 9788121924535
2. J. L. Mott, A. Kandel & T. P. Baker, Discrete Mathematics for Computer Scientists and Mathematicians, 2nd Edition, Prentice Hall of India, ISBN: 9788120314429

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/

OBJECT ORIENTED PROGRAMMING USING C++

Course Code: 2603MC06

L	T	P	C
3	0	2	5

Course Outcomes:

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

CO1: Understand fundamental concepts of object oriented programming and C++ basics.

CO2: Apply classes, objects, constructors and inheritance to develop C++ programs.

CO3: Implement polymorphism, operator overloading and virtual functions in problem solving.

CO4: Use file handling and exception handling mechanisms in C++ applications.

CO5: Develop small real-world applications using object oriented design principles.

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	PO9	PO10
CO1	3	2	1	-	-	-	-	-	-	-
CO2	3	3	2	1	-	-	-	-	-	-
CO3	1	3	3	2	1	-	-	-	-	-
CO4	-	-	3	3	2	1	-	-	-	-
CO5	-	-	-	3	3	2	2	2	1	-

Mapping of Course Outcomes with Program Specific Outcomes:

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

UNIT – I

Introduction to C++ and OOP Concepts: Introduction to Object Oriented Programming, characteristics and advantages of OOP, comparison of procedural and object-oriented programming, structure of C++ program, tokens in C++, data types, operators, control statements, functions in C++, classes and objects, access specifiers, basic input/output using cin and cout.

Practice:

1. Write a C++ program to use different data types and operators.
2. Write programs using if–else and switch statements.
3. Create a class and display student information using objects.

4. Write a C++ program using functions to perform arithmetic operations.
5. Develop a menu-driven program using control statements.

UNIT – II

Constructors, Destructors and Inheritance: Constructors and destructors, types of constructors (default, parameterized, copy), static data members and member functions, friend function, inline function, concept of inheritance, types of inheritance, base and derived classes, constructor invocation in inheritance, real-world inheritance examples.

Practice:

1. Write a C++ program using default and parameterized constructors.
2. Implement a copy constructor in a class.
3. Develop a program using static data members.
4. Write programs illustrating single and multilevel inheritance.
5. Create a class hierarchy for employee management using inheritance.

UNIT – III

Polymorphism and Operator Overloading: Function overloading, operator overloading, rules for operator overloading, virtual functions, pointers to objects, runtime polymorphism, abstract classes and pure virtual functions, introduction to templates and generic programming, function and class templates, basics of namespaces.

Practice:

1. Write a C++ program to implement function overloading.
2. Write a C++ program to implement operator overloading for arithmetic operations.
3. Write a C++ program using virtual functions for runtime polymorphism.
4. Write a C++ program using a template function for sorting elements.
5. Write a C++ program to create and use a simple abstract class.

UNIT – IV

File Handling and Exception Handling: Streams in C++, file handling operations, opening and closing files, file modes, reading and writing text and binary files, random file access, file pointers and file positioning functions, basics of exception handling, try–catch blocks, multiple catch statements, user-defined exceptions, stack unwinding and standard exceptions.

Practice:

1. Write a program to write student data to a file.
2. Develop a program to read and display file contents.
3. Write a C++ program to perform binary file operations.
4. Write programs using try–catch blocks.
5. Implement a user-defined exception handling program.

UNIT – V

Standard Template Library and Applications: Introduction to Standard Template Library (STL), overview of STL components, containers (vector, list, map, set), iterators and their types, STL algorithms (sorting, searching, and numeric algorithms), basics of string class, introduction to simple GUI/console based applications, real-world application development using OOP concepts and STL.

Practice:

1. Write programs using vector and list containers.
2. Write a C++ program to perform sorting using STL algorithms.
3. Develop programs using map container.
4. Implement string manipulation using C++ string class.
5. Create a mini project such as a student record management system.

Additional Practice:

1. Write a program to develop a Student Management System using classes and objects to store student details and find the highest marks.
2. Create a Bank Account system using OOP concepts to perform deposit and withdrawal operations and display the final balance.
3. Develop a Library Management System using inheritance to manage book and issued book information.
4. Write a program to implement an Employee Salary System using polymorphism for different categories of employees.

Textbooks:

1. E. Balagurusamy, Object Oriented Programming with C++, 8th Edition, McGraw Hill Education, ISBN: 978-9353165129
2. Yashavant Kanetkar, Object Oriented Programming with C++, 4th Edition, BPB Publications, ISBN: 978-8183331636

Reference Books:

1. Bjarne Stroustrup, The C++ Programming Language, 4th Edition, Addison-Wesley Professional, ISBN: 978-0321563842
2. Ashok N. Kamthane, Object Oriented Programming with C++, 2nd Edition, Pearson Education, ISBN: 978-8131709821

Web Links:

1. https://www.cplusplus.com
2. https://www.learncpp.com

SOFTWARE ENGINEERING

Course Code: 2603MC07

L	T	P	C
3	0	0	3

Course Outcomes:

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

CO1: Understand the fundamentals, types, applications, and evolving nature of software and software engineering.

CO2: Compare traditional and agile software process models, including Extreme Programming.

CO3: Analyze functional and non-functional software requirements by eliciting, specifying, validating, and managing them.

CO4: Design software architectures and models using design principles and UML diagrams.

CO5: Evaluate software quality using appropriate testing techniques, debugging methods, and metrics.

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	PO9	PO10
CO1	3	2	-	-	-	-	-	-	-	2
CO2	2	3	2	1	-	-	-	-	-	2
CO3	2	2	1	1	-	2	-	-	-	-
CO4	2	1	3	2	1	-	-	-	2	-
CO5	2	2	2	1	-	-	-	-	-	-

Mapping of Course Outcomes with Program Specific Outcomes:

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

UNIT – I

Introduction to Software Engineering: Software, types of software, characteristics of software, the evolving role of software, changing nature of software, software myths, Process models: The waterfall model, incremental process models, evolutionary process models, the unified process. Agile software development, Extreme programming.

UNIT – II

Software Requirements: Functional and non-functional requirements, user requirements, system requirements, interface specification, the software requirements document. Requirements engineering process: Feasibility studies, requirements elicitation and analysis, requirements validation, requirements management.

UNIT – III

Software Project Planning and Risk Management: Project planning- Software pricing, Plan-driven development, Project scheduling, Agile planning, Estimation techniques.

Risk management: Reactive Vs proactive risk strategies, software risks, risk identification, risk projection, risk refinement, RMMM, RMMM plan.

UNIT – IV

Software Design: Design process and design quality, design concepts, the design model,Basic structural modeling, conceptual model of UML, class diagrams, sequence diagrams, collaboration diagrams, use case diagrams, component diagrams.

Practices:

1. Identified scenarios, find the interaction between objects and represent them using UML Sequence and Collaboration Diagrams.
2. Draw relevant State Chart and Activity Diagrams for the same system.

UNIT – V

Software Testing and Quality Management: A strategic approach to software testing, black-box and white-box testing, validation testing, system testing, the art of debugging. Product metrics: Software quality, metrics for analysis model.

Quality Management: Quality concepts, software quality assurance, software reviews, formal technical reviews, statistical software quality assurance, software reliability, the ISO 9000 quality standards.

Textbooks:

1. Roger S. Pressman, Software Engineering: A Practitioner's Approach, 6th Edition, McGraw-Hill, ISBN-13: 9780070701137
2. Sommerville, Software Engineering, 7th Edition, Pearson Education, ISBN-13: 9780321210265

Reference Books:

1. N. S. Gill, Software Engineering, Khanna Publishing House, ISBN-13: 9788190611633
2. Meiler page- Fundamentals of object-oriented design using UML Jones: Pearson Education, ISBN-13: 9780201699463

Web Links:

1. https://www.cplusplus.com
2. https://www.learncpp.com

DATA STRUCTURES

Course Code: 2603MC08

L	T	P	C
2	0	2	4

Course Outcomes:

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

CO1: Understand the fundamental concepts of data structures and their classifications.

CO2: Implement stack and queue operations using arrays and linked lists.

CO3: Apply tree structures for efficient data storage and retrieval.

CO4: Analyze graph-based algorithms and their applications.

CO5: Implement sorting and searching techniques for data organization.

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	PO9	PO10
CO1	3	2	1	1	-	-	-	-	-	2
CO2	2	3	3	2	1	-	-	-	-	2
CO3	2	3	3	2	-	-	-	-	-	2
CO4	2	3	2	2	-	-	-	-	-	2
CO5	2	3	2	2	-	-	-	-	-	2

Mapping of Course Outcomes with Program Specific Outcomes:

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

UNIT – I

Introduction to Data Structures: Abstract Data Types (ADTs) - Data Types, Data Structures, Primitive and Non-primitive Data Structures, Linear and Non-linear Structures. Linear Lists - ADT, Array and Linked representations (Single and Double Linked lists), Circular Linked Lists.

Practice:

1. Write a program to insert an element into a Singly Linked List: (a) At the beginning (b) At the end (c) At a specified position
2. Write a program to delete an element from a Singly Linked List: (a) At the beginning (b) At the end (c) A specified element

3. Write a program to insert an element into a Doubly Linked List: (a) At the beginning (b) At the end (c) At a specified position
4. Write a program to delete an element from a Doubly Linked List: (a) At the beginning (b) At the end (c) A specified element

UNIT – II

Stacks and Queues: Stacks: Definition, Stacks using Array and Linked representations, expressions, notations. Queues: Definition, Queue using Array and Linked representations, Circular Queues, Dequeues.

Practice:

1. Write a program to implement stack operations using an array.
2. Write a program to evaluate a postfix expression using a stack.
3. Write a program to implement queue operations using an array.
4. Write a program to implement circular queue operations using an array.

UNIT – III

Trees: Binary Tree: Definition, Properties, Trees using Array and Linked representations, Implementations and Applications. Binary Search Trees (BST): Definition, Operations and Implementations.

Practice:

1. Write a program to perform the following operations on a Binary search tree. (a) Preorder Traversal (b) Inorder Traversal (c) Postorder Traversal.
2. Implement and analyze operations on Binary Search Trees

UNIT – IV

Graphs: Graph and its Representation, Graph Traversals, Connected Components, Minimum cost spanning Trees, shortest path.

Practice:

1. Implement DFS and BFS graph traversal algorithm.
2. Implement Kruskal's algorithm for finding Minimum Spanning Tree.
3. Implement Dijkstra's Algorithm for finding shortest path.

UNIT – V

Sorting and Searching: Selection Sort, Insertion Sort, Bubble Sort, Merge Sort, Quick Sort, Linear and Binary Searching.

Practice:

1. Write a program to search for an element in an array using Linear Search and Binary Search.
2. Write a program to sort an array using Bubble Sort, Selection Sort and Insertion Sort.
3. Write a program to sort an array using Quick Sort, Merge Sort.

Additional Practice:

1. Write a C program to implement Stack using Linked List and perform the operations.
2. Write a C program to implement Circular Queue using array and perform insertion, deletion and display operations.

Textbooks:

1. Ellis Horowitz, Sartaj Sahni, Susan Anderson-Freed, Fundamentals of Data Structures in C, 2nd Edition, Universities Press, ISBN: 978-8173716058.
2. Mark Allen Weiss, Data Structures and Algorithm Analysis in C, 2nd Edition, Pearson, ISBN: 978-8131703284.

Reference Books:

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, Introduction to Algorithms, 3rd Edition, MIT Press, ISBN: 978-0262033848
2. Reema Thareja, Data Structures Using C, 2nd Edition, Oxford University Press, ISBN: 978-0198099307

Web Links:

1. https://nptel.ac.in/courses/106/102/106102064
2. https://www.geeksforgeeks.org/data-structures
3. https://www.w3schools.com/ds/ds_algorithms_sorting.asp

OPERATING SYSTEMS

Course Code: 2603MC09

L	T	P	C
2	0	1	3

Course Outcomes:

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

CO1: Explain the structure, functions, and services of modern operating systems.

CO2: Analyze process management, threading, and CPU scheduling techniques.

CO3: Apply synchronization mechanisms and handle deadlocks in concurrent systems.

CO4: Evaluate different memory management and virtual memory techniques.

CO5: Use basic Linux commands and shell scripting for file and process management

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	PO9	PO10
CO1	3	2	1	-	-	-	-	-	-	-
CO2	1	3	2	1	-	-	-	-	-	-
CO3	1	2	3	-	-	-	-	-	-	-
CO4	2	1	1	3	-	-	-	-	-	-
CO5	1	1	2	2	-	-	1	-	-	-

Mapping of Course Outcomes with Program Specific Outcomes:

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

UNIT – I

Introduction to Operating Systems & System Structures: Definition and functions of Operating Systems, Types of Operating Systems, Abstract view of OS, Operating system structures, System calls and services, Virtual machines.

Practice:

1. Write a C program to demonstrate basic system calls such as getpid(), getppid(), and system() to display process and system information.
2. Write a C program using fork() to create a child process and show the execution of parent and child processes with their respective process IDs.

UNIT – II

Process Management & Multithreading: Process concept and process states, Process control block, Threads and multithreading models, Inter-process communication (IPC), Process coordination

Practice:

1. Write a C program to implement inter-process communication using pipes where the parent process sends a message to the child process.
2. Write a multithreaded program using POSIX threads (pthreads) to perform concurrent tasks such as calculating the sum and average of an array

UNIT – III

CPU Scheduling & Synchronization: CPU scheduling concepts, Scheduling criteria, Scheduling algorithms, Process synchronization, Critical section problem, Semaphores and monitors, Hardware synchronization mechanisms, Deadlocks.

Practice:

1. Write a program to simulate CPU scheduling algorithms such as FCFS and SJF, and compute waiting time and turnaround time for each process.
2. Write a program to solve the Producer–Consumer problem using semaphores or mutex locks to achieve process synchronization.

UNIT – IV

Memory Management: Memory management strategies, Contiguous and non-contiguous allocation, Paging and segmentation, Virtual memory, Demand paging, Page placement and page replacement policies.

Practice:

1. Write a program to implement the FIFO page replacement algorithm and calculate the number of page faults for a given reference string.
2. Write a program to implement the LRU page replacement algorithm and compare its performance with FIFO in terms of page faults.

UNIT – V

LINUX Operating System: Introduction to Linux, Features and architecture of Linux, Basic Linux commands, File and directory management commands, Process management commands, File permissions and ownership, Shell basics and simple shell scripting, Linux file system overview**.**

Practice:

1. Execute basic Linux commands for file and directory operations (ls, mkdir, rm, cp, mv).
2. Write a shell script to demonstrate arithmetic operations and conditional statements.
3. Write a shell script to display system information (date, users logged in, current directory).

Textbooks:

1. Silberschatz, A., Galvin, P. B., and Gagne, G., Operating System Concepts, 10th Edition, John Wiley & Sons, 2018, ISBN: 978-1-119-32091-3.
2. Stallings, W., Operating Systems: Internals and Design Principles, 9th Edition, Pearson Education, 2017, ISBN: 978-01-346-7095-9.

Reference Books:

1. Arpaci-Dusseau, R. H., and Arpaci-Dusseau, A. C., Operating Systems: Three Easy Pieces, 1st Edition, Arpaci-Dusseau Books, 2018, ISBN: 978-1-985-08659-3.
2. Tanenbaum, A. S., and Bos, H., Modern Operating Systems, 4th Edition, Pearson Education, 2014, ISBN: 978-01-335-9162-0.

Web Links:

1. https://nptel.ac.in/courses/106102132
2. https://onlinecourses.swayam2.ac.in/cec26_cs07/preview

PROBABILITY AND STATISTICS

Course Code: 2603MC10

L	T	P	C
2	1	0	3

Course Outcomes:

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

CO1: Differentiate between primary and secondary data

CO2: Find the measures of central tendency and Dispersion

CO3: Find the probability of various experiments.

CO4: Apply various probability distributions for both discrete and continuous random variables.

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	PO9	PO10
CO1	3	2	-	-	-	-	-	-	-	-
CO2	3	2	-	-	-	-	-	-	-	-
CO3	3	2	-	-	-	-	-	-	-	-
CO4	3	2	-	-	-	-	-	-	-	-
CO5	3	2	-	-	-	-	-	-	-	-

UNIT – I

Introduction: Statistical Data and Methods, Applications of Statistics, Primary and Secondary data, Methods of collecting primary data, Tabulation of data, Diagrammatic and Graphic presentation of data.

UNIT – II

Measures of Central Tendency & Dispersion: Mean, Median, Mode, Range, Mean deviation, Standard deviation.

UNIT – III

Probability: Probability, Axioms of Probability, Addition theorem on probability, Conditional event, Conditional Probability, Multiplication theorem on probability.

UNIT – IV

Probability Distributions: Random variable and probability distribution, Binomial distribution, fitting a Binomial distribution, Poisson distribution, Fitting a Poisson distribution, Normal distribution, Area under the normal curve.

UNIT – V

Correlation and Regression: Types of correlation, Methods of studying Correlation, Karl pearson's coefficient of Correlation, Properties of the coefficient of correlation, Rank correlation coefficient, Uses of regression analysis, Difference between correlation and regression analysis, Regression lines, Regression equations.

Textbooks:

1. S. P. Gupta, Statistical Methods, 46th Revised Edition, Sultan Chand & Sons, ISBN: 9789351611936
2. T. B. Veerajan, Probability, Statistics and Random Processes, 3rd Edition, Tata McGraw Hill Education, ISBN: 9780070145894

Reference Books:

1. Richard A. Johnson (Miller & Freund), Probability and Statistics for Engineers, 7th Edition, Pearson Education, ISBN: 9788131717462
2. T. K. V. Iyengar, B. Krishna Gandhi, S. Ranganatham & M. V. S. S. N. Prasad, Probability and Statistics, 1st Edition, S. Chand Publications, ISBN: 9788121924511

Web Links:

1. nptel.ac.in/courses/111106112
2. nptel.ac.in/courses/110106064

WEB TECHNOLOGIES

Course Code: 2603MC11

L	T	P	C
1	0	2	3

Course Outcomes:

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

CO1: Understand and apply HTML concepts to design well-structured static web pages.

CO2: Apply CSS concepts to style web pages and build responsive layouts.

CO3: Develop dynamic and interactive web pages using JavaScript.

CO4: Build single-page applications using AngularJS framework concepts.

CO5: Develop server-side applications and RESTful services using Node.js.

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	PO9	PO10
CO1	3	2	1	1	1	1	1	1	-	-
CO2	3	2	3	2	1	1	1	1	-	-
CO3	2	3	2	2	1	1	1	1	-	-
CO4	2	2	2	3	1	1	1	1	-	-
CO5	2	2	3	3	2	2	2	1	1	1

Mapping of Course Outcomes with Program Specific Outcomes:

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

UNIT – I

HTML: Introduction to Web & Internet, HTML basic syntax and document structure, Text formatting tags ,Images, audio, video ,Hyperlinks and navigation ,Lists (ordered, unordered, description), Tables and table formatting, Forms and form elements, GET and POST methods (conceptual), HTML5 features: Semantic elements, Audio & video, Forms enhancements ,Input types and attributes.

Practice:

1. Design a simple personal profile page using headings, paragraphs, images, and links.
2. Create a registration form using HTML5 input types and validation attributes.
3. Design a college website homepage using semantic HTML5 elements.

UNIT – II

CSS: Introduction to CSS, Types of CSS: Inline, Internal, External ,CSS selectors (basic, group, class, id), Box model, Colors, backgrounds, borders, Text and font styling,Positioning (static, relative, absolute), Layout techniques: Flexbox, Basic responsive design concepts, Media queries (introduction), CSS3 features: Transitions, Transformations, Animations.

Practice:

1. Apply external CSS to style an HTML page.
2. Design a responsive webpage layout using Flexbox.
3. Create a navigation menu with hover effects using CSS.

UNIT – III

JavaScript: Introduction to JavaScript, Variables, data types, operators, Control structures (if, switch, loops), Functions and arrays, Objects and built-in objects, DOM manipulation, Events (mouse and keyboard), Form validation using JavaScript, JavaScript ES6 basics: let & const, Arrow functions, Introduction to JSON.

Practice:

1. Write a JavaScript program to validate a login form**.**
2. Create a program to change webpage content dynamically using DOM.
3. Develop a JavaScript program to handle mouse and keyboard events**.**

UNIT – IV

AngularJS: Introduction to AngularJS, MVC architecture, AngularJS expressions, Modules and controllers, Data binding, Directives, Filters, Forms and validation, Services and dependency injection, Routing (single-page application concept).

Practice:

1. Create a single-page application using AngularJS.
2. Develop a student details app using controllers and data binding.
3. Implement form validation using AngularJS.

UNIT – V

Node.js: Introduction to Node.js, Advantages of Node.js, Node.js architecture and event loop, Modules (core and custom), File system module, HTTP module, Express.js basics, REST API introduction, Connecting Node.js with MongoDB (conceptual)

Practice:

1. Write a Node.js program to read and write files.
2. Create a simple web server using Node.js.
3. Develop a basic REST API using Express.js.

Additional Practice:

1. Design and develop a responsive personal portfolio website using HTML and CSS.
2. Develop a dynamic form with client-side validation using JavaScript.
3. Create a single-page application using AngularJS for displaying records.
4. Develop a CRUD application using Node.js and Express.js.
5. Implement a RESTful API and test it using Postman.

Textbooks:

1. Robert W. Sebesta, Programming the World Wide Web, 8th Edition, Pearson Education, ISBN: 978-0133775983.
2. Uttam K. Roy, Web Technologies, 2nd Edition, Oxford University Press, ISBN: 978-0199456634.

Reference Books:

1. David Sawyer McFarland, JavaScript & jQuery: The Missing Manual, 3rd Edition, O'Reilly Media, ISBN: 978-1491947074.
2. Dreamtech Press, Web Technologies: HTML, JavaScript, PHP, Java, JSP, XML and AJAX, 1st Edition, Wiley India, ISBN: 978-8126554800.

Web Links:

1. https://developer.mozilla.org
2. https://nptel.ac.in

COMPUTER ARCHITECTURE

Course Code: 2603MC12

L	T	P	C
3	0	0	3

Course Outcomes:

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

CO1: Explain functional units and data representation in computer systems.

CO2: Analyze instruction set architecture and CPU organization.

CO3: Describe memory hierarchy and memory management techniques.

CO4: Understand input–output organization and data transfer mechanisms.

CO5: Evaluate pipelining and parallel processing concepts.

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	PO9	PO10
CO1	3	2	-	-	-	-	-	-	-	1
CO2	3	3	2	-	-	-	-	-	-	1
CO3	3	3	2	-	-	-	-	-	-	1
CO4	2	2	-	2	-	-	-	-	-	1
CO5	3	3	2	1	-	-	-	-	-	2

Mapping of Course Outcomes with Program Specific Outcomes:

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

UNIT – I

Basic Computer Structure and Data Representation: Functional units of a computer system, Von Neumann architecture, register transfer language, instruction cycle, bus structure and performance. Data representation including signed numbers, complements, fixed and floating-point representation. Computer arithmetic: addition, subtraction, ripple carry adder, carry look-ahead adder, multiplication algorithms, division algorithms and floating-point arithmetic.

UNIT – II

Instruction Set and CPU Organization: Instruction set architecture, instruction formats, addressing modes, instruction sequencing, CISC vs RISC architecture. Basic processing unit design, execution of instructions, data path organization, hardwired and microprogrammed control, and multiple-bus organization.

UNIT – III

Memory Organization: Semiconductor memory technologies, RAM, ROM, memory characteristics, main memory and auxiliary storage. Memory hierarchy design, cache organization, mapping techniques, write policies, memory interleaving, virtual memory and paging, and memory management requirements

UNIT – IV

Input–Output Organization: I/O interfaces and peripheral devices, program-controlled I/O, interrupt-driven I/O, DMA, interrupts and exceptions, asynchronous transfer, priority interrupt systems, I/O buses and interface circuits, and standard I/O interfaces such as USB and SCSI.

UNIT – V

Pipelining and Parallel Processing: Instruction and arithmetic pipelining, pipeline hazards and hazard handling techniques, superscalar operations, vector and array processing, shared memory multiprocessors, interconnection networks, cache coherence, and multiprocessor communication.

Textbooks:

1. M. Morris Mano, Computer System Architecture, Third Edition, Pearson Education, ISBN: 978-0131755635
2. William Stallings, Computer Organization and Architecture, Tenth Edition, Pearson Education, ISBN: 978-0134101613

Reference Books:

1. Carl Hamacher, Zvonko Vranesic, Safwat Zaky, Computer Organization, Fifth Edition, McGraw-Hill Education, ISBN: 978-0070660038
2. Andrew S. Tanenbaum, Todd Austin, Structured Computer Organization, Sixth Edition, Pearson Education, ISBN: 978-0132916523

Web Links:

1. https://www.nptel.ac.in/courses/106106166
2. https://www.geeksforgeeks.org/computer-organization-architecture/computer-organization-and-architecture-tutorials/

JAVA PROGRAMMING

Course Code: 2603MC13

L	T	P	C
2	0	2	4

Course Outcomes:

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

CO1: Understand basic OOP concepts and the structure and features of Java programs.

CO2: Apply classes, objects, and methods to develop Java applications.

CO3: Use inheritance, interfaces, and packages in Java programs.

CO4: Implement exception handling and multithreading in Java.

CO5: Develop database applications using JDBC and perform CRUD 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	PO9	PO10
CO1	3	2	-	-	-	-	-	-	-	1
CO2	2	3	1	-	-	-	-	-	-	1
CO3	2	2	2	-	-	-	-	-	-	1
CO4	1	1	2	1	1	-	-	-	-	1
CO5	2	2	2	2	1	-	-	1	-	1

Mapping of Course Outcomes with Program Specific Outcomes:

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

UNIT – I

Introduction to OOP and Java Fundamentals: Introduction to Object-Oriented Programming concepts, Overview and features of Java, Structure of a Java program, Compilation and execution of a simple Java program, Java tokens (keywords, identifiers, literals, operators, separators), Data types, Variables, Type conversion and casting, Operators in Java, Command line arguments, Conditional statements (if, if–else, switch), Looping statements (for, while, do–while).

Practice:

1. Write a program to demonstrate command line arguments and perform arithmetic operations.
2. Write a program to find the largest of three numbers using conditional statements and display a multiplication table using loops.
3. Write a java Program to find given number is Armstrong or not

UNIT – II

Classes, Objects, and Methods: Introduction to classes and objects, Class declaration and modifiers, Object creation using new, Method definition, Method overloading, The this keyword, Constructors and constructor overloading, Access control (public, private, protected, default), Static and final members, Nested classes.

Practice:

1. Write a Java program to create a class Student with data members and methods to display details using objects.
2. Write a Java program to illustrate method overloading.
3. Write a program to demonstrate constructor overloading in a class Rectangle to calculate area.
4. Write a Java program to demonstrate the use of static and final keywords.

UNIT – III

Inheritance, Interfaces and Packages: Introduction to inheritance, Types of inheritance, super keyword, Method overriding, Dynamic method dispatch, Abstract classes, Introduction to interfaces, Declaring and implementing interfaces, Multiple interfaces, Interface inheritance, Introduction to packages, Java built-in packages, Creating and importing packages.

Practice:

1. Write a program to demonstrate single inheritance using a class Person and derived class Employee.
2. Write a program to implement an interface Shape with methods to calculate area of circle and rectangle.
3. Create a user-defined package and import it into another program to access class members.

UNIT – IV

Exception Handling and Multithreading: Introduction to exception handling, Exception hierarchy, try–catch–finally blocks, throw and throws keywords, Multiple catch blocks, Checked and unchecked exceptions, Introduction to multithreading, Thread life cycle, Thread priorities, Runnable interface and Thread class, Thread methods (start(), sleep(), join(), yield()), Thread synchronization, Inter-thread communication.

Practice:

1. Write a Java program to handle division by zero using try–catch blocks.
2. Write a Java program demonstrating multiple catch blocks.
3. Write a Java program to create threads using both Thread class and Runnable interface.
4. Write a Java program to demonstrate thread synchronization.

UNIT – V

Introduction to JDBC: Overview of JDBC, JDBC architecture, Types of JDBC drivers, Steps to connect to a database, Loading the driver, Establishing connection using DriverManager, Creating and executing SQL statements using Statement and PreparedStatement, Handling ResultSet, Performing CRUD operations.

Practice:

1. Write a Java program to establish a JDBC connection and display database metadata.
2. Write a Java program to perform CRUD operations (insert, select, update, delete) on a student table using JDBC.
3. Write a Java program to retrieve and display records using PreparedStatement

Additional Practice:

1. Develop a Java program to show dynamic method dispatch using superclass reference.
2. Write a Java program to demonstrate inter-thread communication using wait() and notify().
3. Develop a program to handle NumberFormatException and ArithmeticException using multiple catch blocks.
4. Write a Java program to count the number of records in a table using JDBC.

Textbooks:

1. JAVA one step ahead, Anitha Seth, B.L.Juneja, Oxford, ISBN-13: 978-0199459643.
2. Joy with JAVA, Fundamentals of Object Oriented Programming, DebasisSamanta,
3. MonalisaSarma, Cambridge, 2023, ISBN-13: 978-1009211918.
4. JAVA 9 for Programmers, Paul Deitel, Harvey Deitel, 4th Edition, Pearson, ISBN-13: 978-9352866540.

Reference Books:

1. The complete Reference Java, 11thedition, Herbert Schildt,TMH,ISBN-10: 1260440249.
2. Introduction to Java programming, 7th Edition, Y Daniel Liang, Pearson, ISBN-10: 8131729583.

Web Links:

1. https://docs.oracle.com/javase/tutorial/.
2. https://www.javatpoint.com/java-tutorial.

ARTIFICIAL INTELLIGENCE

Course Code: 2603MC14

L	T	P	C
2	0	2	4

Course Outcomes:

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

CO1: Understand Artificial Intelligence concepts, intelligent behavior, and problem characteristics.

CO2: Apply uninformed and informed search techniques to solve simple AI problems.

CO3: Represent knowledge using logical and rule-based approaches and perform basic reasoning.

CO4: Explain expert systems and uncertainty handling techniques used in AI.

CO5: Understand Natural Language Processing concepts and AI 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	PO9	PO10
CO1	3	2	-	-	-	-	-	1	-	-
CO2	3	2	2	1	-	-	-	1	-	-
CO3	2	2	2	-	-	-	-	-	-	-
CO4	3	1	2	1	-	-	1	-	-	-
CO5	-	2	2	1	1	-	1	1	-	-

Mapping of Course Outcomes with Program Specific Outcomes:

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

UNIT – I

Introduction to Artificial Intelligence: Definition and scope of Artificial Intelligence. Brief history and development of Artificial Intelligence. Applications of Artificial Intelligence in areas such as healthcare, education, and banking. Comparison between human intelligence and machine intelligence. Basic idea of problem solving using simple examples.

Practice:

1. Illustrate Artificial Intelligence using suitable real-life examples.
2. Demonstrate applications of Artificial Intelligence in different fields.

3. Show how a simple problem can be represented step by step.

UNIT – II

Search Techniques for Problem Solving: Introduction to problem solving in Artificial Intelligence. Concept of search techniques for finding solutions. Breadth First Search and Depth First Search with illustrative examples and simple comparison. Introduction to heuristic search and illustration of A* search using a suitable example.

Practice:

1. Demonstrate searching for a solution using a simple example.
2. Illustrate Breadth First Search and Depth First Search using examples.
3. Show the working of A* search with a suitable example.

UNIT – III

Knowledge Representation and Reasoning: Need for knowledge representation in Artificial Intelligence. Types of knowledge with examples. Rule-based knowledge representation using IF–THEN rules. Introduction to logical statements used for simple decision making.

Practice:

1. Illustrate knowledge representation using suitable examples.
2. Construct IF–THEN rules for simple real-life situations.
3. Demonstrate logical statements using examples.

UNIT – IV

Expert Systems: Introduction to expert systems and their applications. Basic components of expert systems including knowledge base and inference mechanism. Working principle of expert systems using suitable examples. Decision making under uncertainty using simple situations.

Practice:

1. Illustrate an expert system using a real-life application.
2. Demonstrate how an expert system provides solutions.
3. Show decision making using suitable examples.

UNIT – V

Natural Language Processing: Introduction to Natural Language Processing. Processing of human language by computers. Applications such as chatbots, translation systems, and voice assistants. Advantages and limitations of Artificial Intelligence.

Practice:

1. Demonstrate how a chatbot processes user input.
2. Illustrate applications of Natural Language Processing.
3. Show examples of advantages and limitations of Artificial Intelligence.

Additional Practice:

1. Demonstrate an Artificial Intelligence application used in daily life.
2. Illustrate the use of Artificial Intelligence in online platforms.
3. Show step-by-step solution of a simple problem using Artificial Intelligence concepts.
4. Construct rule-based decisions for real-life situations.
5. Present a short case study on any Artificial Intelligence application.

Textbooks:

1. Deepak Khemani, Artificial Intelligence, First Edition, Tata McGraw-Hill Education, ISBN: 978-1259029981
2. Elaine Rich and Kevin Knight, Artificial Intelligence, Third Edition, McGraw-Hill Publications, ISBN: 978-0070087705

Reference Books:

1. Stuart Russell and Peter Norvig, Artificial Intelligence: A Modern Approach, Third Edition, Pearson Education, ISBN: 978-0136042594.
2. George F. Luger, Artificial Intelligence: Structures and Strategies for Complex Problem Solving, Fifth Edition, Pearson Education, ISBN: 978-0321545893.

Web Links:

1. https://nptel.ac.in/courses/106/105/
2. https://www.tutorialspoint.com/artificial_intelligence/
3. https://www.geeksforgeeks.org/artificial-intelligence/

PYTHON PROGRAMMING

Course Code: 2603MC15

L	T	P	C
3	0	2	5

Course Outcomes:

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

CO1: Write basic Python programs using variables, operators, conditions and loops.

CO2: Use lists, tuples, dictionaries and sets for simple data handling.

CO3: Perform string processing and basic pattern matching.

CO4: Develop modular programs using functions with basic exception handling.

CO5: Build small applications using object-oriented programming and file storage.

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	PO9	PO10
CO1	2	2	3	-	2	-	-	1	-	1
CO2	2	2	3	-	2	-	-	1	-	1
CO3	3	-	2	-	2	-	-	2	1	1
CO4	2	1	2	-	2	-	-	1	1	1
CO5	1	1	2	-	2	-	-	2	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

Introduction to Python and Flow Control: Brief history and features of Python, Running Python programs using IDE/interactive shell, Structure of a Python program, Keywords and identifiers, Variables and basic data types: int, float, complex, str, Basic input and output, Operators and simple expressions, Conditional statements: if, elif, else, Loops: for, while, Loop control statements: break, continue, pass.

Practice:

1. Write a Python program to read two numbers from the user and display their sum.
2. Write a Python program to demonstrate different basic data types (int, float, complex, str).
3. Write a Python program to check whether a given number is even or odd using conditional

statements.

4. Write a Python program to print squares of numbers from 1 to N using a for loop.
5. Write a Python program to demonstrate the use of loop control statements break, continue, and pass.

UNIT – II

Lists, Tuples, Dictionaries, Sets and Basic File Handling: Lists: creation, indexing, slicing, add/remove elements, sorting, Tuples: packing, unpacking, simple usage, Range and basic iteration, Dictionaries: create, add, update, delete, search, Sets: basic operations (union, intersection, difference), Introduction to text file handling: open, read, write, close.

Practice:

1. Write a Python program to create a list, perform indexing and slicing, and add and remove elements from the list.
2. What is a tuple? Write a Python program to demonstrate tuple packing and unpacking.
3. Write a Python program to create a dictionary to store student roll number and name, and perform add, update, delete, and search operations.
4. Explain sets in Python. Write a program to perform union, intersection, and difference operations on two sets.
5. Write a Python program to demonstrate basic file handling by writing data to a text file and reading it back using open(), write(), read(), and close() methods.

UNIT – III

Strings and Introductory Regular Expressions: String operations: concatenation, indexing, slicing, Common string methods (lower, upper, find, replace, split), Length and basic validation of strings, Introduction to regular expressions, Simple pattern matching (digits, words, email format)

Practice:

1. Reverse-vowels-of-a-String
2. Count vowels, consonants, digits in a string.
3. Demonstrate major string operations on user input.
4. Validate a simple email format using regular expressions.

UNIT – IV

Functions and Exception Handling: Defining and calling user-defined functions, Parameters and return values, Default arguments, Lambda functions (basic use), Organising programs using functions, Introduction to exceptions, Using try, except, finally for safe programs.

Practice:

1. Write a program to add two numbers using a user-defined function.
2. Write a program to calculate simple interest using a function with a default argument.
3. Write a program to find the square of a number using a lambda function.
4. Write a program to check whether a number is even or odd using a function.

5. Write a program to perform division of two numbers and handle errors using try and except.

UNIT – V

Object Oriented Programming and Simple Persistence: Creating classes and objects, Instance variables and methods, Constructor (__init__), Basic inheritance, Method overriding (simple polymorphism), Difference between instance, class and static methods (introductory), Storing and loading object data using files (basic object persistence through text/JSON).

Practice:

1. Write a Python program to create a class Student with instance variables roll_no, name, marks and a method to display student details.
2. Write a Python program to demonstrate the use of a constructor (__init__) by initializing object data at the time of object creation.
3. Write a Python program to demonstrate the difference between instance variables and class variables using a simple class.
4. Write a Python program to demonstrate single inheritance, where a derived class inherits properties of a base class.
5. Write a Python program to store and retrieve object data using a text file or JSON file

Additional Practice:

1. Write a Python program to check whether a given number is prime or not.
2. Write a program to swap two numbers without using a third variable.
3. Write a Python program to find the factorial of a number using a loop.
4. Write a program to generate the Fibonacci series up to N terms.
5. Write a Python program to check whether a given number is an Armstrong number.
6. Write a program to find the largest and smallest elements in a list.
7. Write a Python program to count the frequency of elements in a list using a dictionary.
8. Write a program to reverse a string and check whether it is a palindrome.
9. Write a Python program to save and load object data using a JSON file.

Textbooks:

1. Kenneth A. Lambert, Fundamentals of Python: First Programs, 2nd Edition, Cengage Learning, ISBN: 978-1305964764.
2. Magnus Lie Hetland, Beginning Python: From Novice to Professional, 2nd Edition, Apress, ISBN: 978-1430224150.

Reference Books:

1. Allen B. Downey, Think Python: How to Think Like a Computer Scientist, 2nd Edition, O'Reilly Media, ISBN: 978-1491939369.
2. Vamsi Kurama, Python Programming: A Modern Approach, 1st Edition, Pearson Education, ISBN: 978-9332585342.

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/

DESIGN AND ANALYSIS OF ALGORITHMS

Course Code: 2603MC16

L	T	P	C
3	1	0	4

Course Outcomes:

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

CO1: Explain the fundamentals of algorithmic problem solving, data structures, and analyze algorithm efficiency using asymptotic notations and mathematical techniques.

CO2: Apply brute-force, divide-and-conquer, and decrease-and-conquer strategies to design and analyze sorting, searching, and graph traversal algorithms.

CO3: Analyze and compare transform-and-conquer techniques, heap structures, and problem reduction methods to solve computational problems efficiently.

CO4: Design algorithms using space–time tradeoffs and dynamic programming techniques for solving optimization problems.

CO5: Evaluate the limitations of algorithmic approaches and implement greedy, backtracking, and branch-and-bound techniques for complex real-world problems

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	PO9	PO10
CO1	3	2	1	1	-	-	-	-	-	-
CO2	-	3	2	1	-	-	-	-	-	-
CO3	1	2	3	1	-	-	-	-	-	-
CO4	2	1	2	1	-	-	-	-	-	-
CO5	1	-	3	1	-	-	-	-	-	-

Mapping of Course Outcomes with Program Specific Outcomes:

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

UNIT – I

Introduction: Fundamentals of algorithmic problem solving, important problem types, fundamental data structures. Fundamentals of analysis of algorithms and efficiency: Analysis framework, Asymptotic Notations and Basic Efficiency classes, Mathematical Analysis of Non-

recursive Algorithms, Mathematical Analysis of recursive Algorithms, Empirical Analysis of Algorithms, Algorithm Visualization.

UNIT – II

Brute Force: Selection Sort and Bubble sort, Sequential Search and Exhaustive Search. **Divide-and-Conquer:** Merge Sort, Quick sort, Binary Search, Binary Tree Traversals and Related Properties. **Decrease-and-Conquer:** Insertion Sort, Depth-First Search and Breadth-First Search-
Topological sorting.

UNIT – III

Decrease-by-a-Constant-Factor Algorithms, Variable-Size-Decrease Algorithms.
Transform-and-Conquer: Presorting, Balanced Search Trees, Heaps and Heap sort, Problem Reduction.

UNIT – IV

Space and Time Tradeoffs: Sorting by Counting, Hashing, B-Trees.Dynamic Programming: Warshall's and Floyd's Algorithm, Optimal Binary Search Trees, The Knapsack Problem and Memory Functions.

UNIT – V

Coping with the Limitations of Algorithms Power: Backtracking, Branchand- Bound
Greedy Technique: Prim's Algorithm, Kruskal's Algorithm, Dijkstra's Algorithm.

Textbooks:

1. Anany Levitin, Introduction to Design & Analysis of Algorithms, Pearson Education, ISBN 13: 978-0-273-76411-3
2. Aho, Hopcroft& Ullman, The Design and Analysis of computer Algorithms, Pearson Education, ISBN-13 978-8178081038

Reference Books:

1. Gilles Brassard & Paul Bratley, Fundamentals of algorithms, Prentice Hall of India, ISBN-13. 978-9332549999
2. Ellis Horowitz, Satra Sahani and Rajasekaran, Fundamentals of Computer Algorithms, ,2008, University Press, ISBN-13. 978-8173716126
3. S. Samantha, Design and Analysis of Algorithms, McGraw Hill Education ISBN: 978-0070143268
4. T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, Introduction to Algorithms, MIT Press, ISBN: 978-0262033848

Web Links:

1. https://www.naukri.com/code360/library/design-and-algorithm-analysis

2. https://ocw.mit.edu/courses/6-046j-design-and-analysis-of-algorithms-spring-2015/pages/lecture-notes/

COMPUTER NETWORKS

Course Code: 2603MC17

L	T	P	C
3	0	2	5

Course Outcomes:

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

CO1: Understand the fundamentals concepts of computer network

CO2: Understand the DLL services and different protocol uses in computer networks

CO3: Understand the uses of various protocols and Connection devices

CO4: Analyze the network layer and transport layer services

CO5: Analyze the application layer and network security in trouble shooting the network

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	PO9	PO10
CO1	2	-	3	-	3	-	-	3	-	-
CO2	2	-	3	-	3	-	-	3	-	-
CO3	2	-	3	1	3	-	-	3	-	-
CO4	2	-	3	1	3	-	-	3	-	-
CO5	-	-	3	1	3	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 Computer Network: Networking Fundamentals: Basics of Networking, Networking Terms- Host, Workstations, Server, Client, Node, Advantages of Networking, Types of Networks, Network Topologies, Types of Transmission Media- Guided and Unguided, Communication Modes. Data communication protocols and standards, Network models –OSI model-layers and their functions, TCP / IP protocol suite.

Practice:

1. Create LAN, MAN, WAN, Client–Server model, Star, Bus, Ring, Mesh topologies
2. Configure IP addressing, Simulate TCP/IP communication,
3. Observe packet flow by OSI layer.

Tools: Cisco Packet Tracer (Cisco Academy) / Wireshark /GNS3 / NS2/NS3.

UNIT – II

Data Link Layer: Data link layer: Error Detection and Correction, Framing, flow and error control, Protocols -Noiseless channels (Simplest, Stop and Wait) and Noisy channels (Stop and Wait and Piggy Backing), PPP.

Practice:

1. Capture Ethernet frames
2. Analyze frame fields (Header, FCS) & Observe CRC checks
3. Identify ARP, PPP, HDLC frames

Tools: Cisco Packet Tracer (Cisco Academy) / Wireshark /GNS3 / NS2/NS3

UNIT – III

Multiple Access Protocols: Multiple Access Protocols, Random Access – ALOHA, CSMA. Connecting Devices - Repeater, Modem, Hub, Switch, Bridge, Router, Gateway. Wired LANs - IEEE standards, wireless LANs - Bluetooth, Cellular Telephony, Satellite Networks, SONET.

Practice:

1. Simulate Ethernet LAN (IEEE 802.3)
2. Observe CSMA/CD collisions
3. Compare Hub vs Switch performance
4. Configure Router & Gateway
5. WLAN (802.11) basic setup
6. Tools: Cisco Packet Tracer (Cisco Academy) / Wireshark /GNS3 / NS2/NS3

UNIT – IV

Network Layer and Transport Layer: Network layer and Transport layer: Logical addressing – IPv4 addressing, IPv4 address Classes, Subnet Mask, Public & Private IP Address and IPV6 addressing, Address mapping-ICMP, IGMP. Connectionless and Connection-Oriented Services: Transport layer services, UDP and TCP. Congestion Control, Quality of Service. Introduction to Routing and Switching concepts.

Practice:

1. Assign IPv4 addresses
2. Practice address classes & subnet masks
3. Configure public & private Ips
4. Basic routing (static, default)
5. Observe TCP vs UDP communication

Tools: Cisco Packet Tracer (Cisco Academy) / Wireshark /GNS3 / NS2/NS3

UNIT – V

Application Layer: Application Layer: DHCP, DNS, HTTP / HTTPS, FTP, TFTP, SFTP, Telnet, Email: SMTP, POP3 / IMAP. Virtual Private Networking, Network security: Common Threats – Firewalls (advantages and disadvantages), Digital Signature, Troubleshooting the network.

Practice:

1. Configure DHCP & DNS servers
2. Simulate HTTP / HTTPS traffic
3. FTP, TFTP server configuration
4. Telnet device access
5. Basic firewall rules
6. Network fault troubleshooting

Tools: Cisco Packet Tracer (Cisco Academy) / Wireshark /GNS3 / NS2/NS3

Additional Practice:

1. Linux / Windows server setup
2. Run multiple services simultaneously
3. VPN server & firewall testing

Tools: Cisco Packet Tracer (Cisco Academy) / Wireshark /GNS3 / NS2/NS3

Textbooks:

1. Kizza, J. M. (2024). Computer Network Security and Cyber Ethics (4th ed.). McFarland & Company, Inc., Publishers. ISBN-13: 978-0-7864-9392-0 / ISBN-10: 0-7864-9392-5
2. Kurose, J. F., & Ross, K. W. (2022/2023). Computer Networking: A Top-Down Approach (8th ed.). Pearson Education. ISBN-13: 978-9356061316/ ISBN-10: 9356061319.
3. Tanenbaum, A. S., Feamster, N., & Wetherall, D. J. (2021). Computer Networks (6th ed.). Pearson Education. ISBN-13: 978-1-292-37406-2.

Reference Books:

1. Tanenbaum, A. S. (2012). Computer Networks (5th ed.). PHI Learning. ISBN-13: 978-0132126953; ISBN-10: 0132126958.
2. Jain, S., Jain, M., Pillai, V., & Kratika. (2010). A Level Data Communication & Network Technologies. BPB Publications. ISBN-13: 978-8183333504.

Web Links:

1. https://www.geeksforgeeks.org/computer-network-tutorials/
2. https://www.javatpoint.com/computer-network-tutorial
3. https://www.tutorialspoint.com/data_communication_computer_network/index.htm
4. https://www.scaler.com/topics/computer-network/

MACHINE LEARNING WITH PYTHON

Course Code: 2603MC18

L	T	P	C
3	0	2	5

Course Outcomes:

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

CO1: Understand Basic Concepts of Machine Learning
CO2: Analyze supervised learning models
CO3: Construct Unsupervised learning algorithms
CO4: Evaluate and compare different ensemble models
CO5: Analyze neural network 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	PO9	PO10
CO1	3	2	1	1	1	1	1	1	-	-
CO2	2	3	1	1	1	1	1	1	-	-
CO3	2	2	1	3	1	1	1	1	-	-
CO4	2	2	3	1	1	1	1	2	-	-
CO5	3	2	2	1	1	1	1	1	-	-

Mapping of Course Outcomes with Program Specific Outcomes:

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

UNIT – I

Introduction to Machine Learning: What is AI? Basic Concepts of Machine Learning, Life cycle of Machine Learning, Types of Machine Learning, Applications of Machine Learning, Well Defined learning problems, Designing a Learning System, Bias and Variance, Major Issues in Machine Learning. **The Concept Learning Task** - General-to-specific ordering of hypotheses, Find-S algorithm

Practice:

1. Write a python program to import and export data using Panda's library functions.
2. Write a python program to implement FIND-S Algorithm?

UNIT – II

Supervised Learning Techniques: Introduction to Classification and Regression Algorithms, Linear Regression, Decision Trees, Random Forest, Naïve bayes, K Nearest Neighbour's, Support Vector Machine.

Practice:

1. Write a python program to implement linear regression using matplotlib.
2. Write a python program to implement decision tree

UNIT – III

Unsupervised Learning Techniques: Introduction to Clustering, Dimensionality Reduction, Principle Component Analysis, K-Means Algorithm, Merits & Demerits of Clustering Techniques

Practice:

1. Implement Dimensionality reduction using Principle Component Analysis (PCA) method.

UNIT – IV

Ensemble Model Learning: Introduction to Ensemble method, Base Learners, Voting Classifier, Bagging, Boosting, Stacking, Overfitting, Underfitting, K-Fold Validation, Limitations.

Practice:

1. Implement Boosting ensemble method on a sample dataset.

UNIT – V

Introduction to Neural Networks: Multi-Layer Perceptron, Gradient Descent, Artificial Neural Network, Activation Function, Optimization, Hyper Parameter Tuning, Regularization, Back Propagation Algorithm.

Practice:

1. Build Artificial Neural Network model with back propagation on a given dataset.

Additional Practice:

1. Write a python program to implement K-Means clustering Algorithm.
2. Implement Naïve Bayes Classification in Python.

Textbooks:

1. Tom Mitchell, "Machine Learning", McGraw Hill, 3rd Edition, ISBN 13: 978-1-292-43749-1.
2. Ian Goodfellow, Yoshua Bengio, Aaron Courville, "Deep Learning", MIT Press, ISBN:9780262035613

Reference Books:

1. Stephen Marsland, “Machine Learning: An Algorithmic Perspective, “Second Edition”, CRC Press, ISBN 9781466583283.
2. Ethem Alpaydin, “Introduction to Machine Learning”, MIT Press, Fourth Edition, ISBN: 9780262043793.

Web Links:

1. nptel.ac.in/courses/106106139
2. https://data-flair.training/blogs/data-mining-tutorial/
3. https://machinelearningmastery.com/

OBJECT ORIENTED SOFTWARE ENGINEERING

Course Code: 2603MC19

L	T	P	C
3	0	1	4

Course Outcomes:

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

CO1: Understand basics of requirements engineering and object-oriented analysis.
CO2: Identify functional and non-functional requirements using basic techniques.
CO3: Use UML diagrams to model software systems.
CO4: Design simple user-centered interfaces and system interactions.
CO5: Understand software design principles and reusability concepts.

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	PO9	PO10
CO1	3	2	1	-	-	-	-	-	-	-
CO2	2	3	2	-	-	-	-	-	-	-
CO3	2	2	3	1	-	-	-	-	-	-
CO4	2	2	3	1	-	-	-	1	-	-
CO5	2	2	3	2	-	-	-	-	1	2

Mapping of Course Outcomes with Program Specific Outcomes:

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

UNIT – I

Introduction to Software Engineering & Requirements: Introduction to Software Engineering, Need for Software Engineering, Software Characteristics, Software Development Life Cycle (SDLC – overview), Introduction to Requirements Engineering, Functional Requirements, Non-Functional Requirements, Importance of Requirements in Software Development.

Practice:

1. For a Gym Management System, list 5 functional and 5 non-functional requirements.
2. Write a short SRS outline for an Online Event Registration System.

UNIT – II

Requirement Gathering & Object-Oriented Analysis: Requirement Gathering Techniques (Interview, Questionnaire, Observation), Introduction to Object-Oriented Analysis (OOA), Basic OOP Concepts (Object, Class, Attributes, Methods), Identifying Objects and Classes from a problem statement, Introduction to Use Cases, Actors and Use Cases, Simple Use Case Diagrams.

Practice:

1. For a Hospital Management System, choose a requirement gathering technique and write 6 interview questions.
2. From an Online Food Ordering System, identify classes, attributes, methods, and actors.

UNIT – III

Unified Modeling Language (UML): Introduction to UML, Importance of UML, UML Diagrams: Use Case Diagram, Activity Diagram, Class Diagram, Class Diagram with Association and Multiplicity, Generalization (Inheritance), Simple UML Case Studies.

Practice:

1. Draw a Use Case and Class Diagram (with multiplicity) for a Course Registration System.
2. Draw an Activity Diagram and State Diagram for an Online Payment Process.

UNIT – IV

User Interface Design & System Behaviour: Introduction to User-Centred Design, Types of Users, Basics of User Interface Design, Usability Principles, Common UI Design Problems, Introduction to Activity Diagram and State Diagram, Modelling simple system behaviour.

Practice:

1. Identify usability problems and redesign a poorly structured Login Page.
2. For a Scholarship Portal, identify user types and draw an Activity Diagram.

UNIT – V

Software Design & Reusability: Introduction to Software Design, Characteristics of Good Design, Basic Design Principles, Association and Aggregation, Introduction to Software Reusability, Overview of Design Patterns (concept only), Introduction to Client–Server Architecture, Role of Object-Oriented Design in Real-World Applications.

Practice:

1. Design a simple Client–Server architecture for an Online Food Delivery System.
2. Apply basic design principles and suggest one design pattern for a Library Management System.

Textbooks:

1. Roger S. Pressman, Software Engineering: A Practitioner's Approach, 8th Edition, McGraw Hill, ISBN: 978-0070708453
2. Ali Bahrami, Object Oriented Systems Development, McGraw Hill, ISBN: 978-0070639927

Reference Books:

1. Ian Sommerville, Software Engineering, 10th Edition, Pearson Education, ISBN: 978-0133943030
2. Grady Booch, James Rumbaugh & Ivar Jacobson, The Unified Modeling Language User Guide, 2nd Edition, Pearson, ISBN: 978-0321267979

Web Links:

1. https://www.tutorialspoint.com/software_engineering
2. https://www.geeksforgeeks.org/software-engineering/

SOFTWARE PROJECT MANAGEMENT

Course Code: 2603MC20

L	T	P	C
3	0	1	4

Course Outcomes:

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

CO1: Evaluate project feasibility and organize projects using the 4P's framework.

CO2: Perform advanced scheduling using PERT/CPM and resource leveling techniques.

CO3: Apply COCOMO II and function point analysis for accurate effort estimation

CO4: Develop comprehensive RMMM plans to mitigate technical and business risks.

CO5: Monitor project health using Earned Value Management (EVM) and reporting tools.

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	PO9	PO10
CO1	2	1	-	-	3	2	-	-	2	1
CO2	3	2	-	-	1	-	-	-	2	-
CO3	2	1	-	1	-	-	-	-	3	-
CO4	-	3	-	-	1	1	-	-	2	-
CO5	1	1	-	2	-	2	-	-	1	2

Mapping of Course Outcomes with Program Specific Outcomes:

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

UNIT – I

Project Organization and the 4p's Framework: Understanding Software Projects and their unique characteristics; Software Project Management vs. General Management; The 4P's for effective Project Management: People (Stakeholders, Team Leader, Software Team, Coordination), Product (Software Scope, Problem Decomposition), Process (Melding Product and Process), and Project (Critical Success Factors); Role and Competencies of a Modern Project Manager; Professional Ethics in Project Management.

Practice:

1. Identify a real-world software project and list the stakeholders involved based on the 4P's framework.
2. Draft a "Role Description" for a Project Manager for a small startup project versus a large MNC project.
3. Analyze a case study of a failed software project and identify which of the 4P's was most significantly mismanaged.

UNIT – II

Selection & Infrastructure: Strategic Project Selection: Technical and Economic Feasibility; Project Infrastructure: Establishing a project environment, Hardware and Software resource planning; Identifying project products and activities; Work Breakdown Structure (WBS); Analyzing project characteristics: Objectives, Scope, and Constraints; Selecting appropriate project methodologies (Beyond standard SDLC).

Practice:

1. Create a Work Breakdown Structure (WBS) for a "Library Management System" down to the task level.
2. Define the project infrastructure (hardware, software tools, and communication channels) for a remote development team.
3. Perform a feasibility study for a proposed mobile application, evaluating technical, economic, and legal constraints.

UNIT – III

Advanced Scheduling: Time management and Scheduling Fundamentals; Network Diagrams: Precedence Diagramming Method (PDM); Critical Path Method (CPM): Forward Pass, Backward Pass, and Float Calculation; Program Evaluation and Review Technique (PERT): Three-point estimation and Probability of completion; Resource Allocation and Leveling; Project Crashing: Time-Cost Trade-offs.

Practice:

1. Draw a PERT chart for 10 project activities and calculate the Expected Time for each using the optimistic, most likely, and pessimistic time estimates.
2. Given a project network, identify the Critical Path and calculate the total float for all non-critical activities.
3. Perform a "Project Crashing" exercise: determine which activities should be shortened to reduce the timeline by 2 weeks with minimum cost

UNIT – IV

Software Economics and Cost Estimation: Decomposition Techniques: Software Sizing (KLOC, Function Point Analysis); Problem-Based vs. Process-Based Estimation; Empirical Estimation Models: Detailed COCOMO II Model; Software Pricing and Budgeting; Automated Estimation Tools; Factors influencing project cost and effort.

Practice:

1. Calculate the Effort and Development Time for a 30 KLOC project using the Basic COCOMO model.
2. Conduct a Function Point (FP) analysis for a simple login and data entry module based on external inputs and outputs.
3. Compare the estimation results of the COCOMO model against the Process-based estimation for the same project size.

UNIT – V

Risk Management and Project Monitoring: Risk Management: Reactive vs. Proactive strategies; Risk Identification: Product, Process, and Business Risks; Risk Assessment: Impact and Probability; Risk Mitigation, Monitoring, and Management (RMMM) Plan; Project Monitoring: Earned Value Management (EVM), Schedule Variance, Cost Variance; Project Closure: Post-mortem analysis and Knowledge Management.

Practice:

1. Develop a Risk Table for a project, ranking 5 potential risks by their probability and impact.
2. Draft a complete RMMM plan for the highest-ranked risk identified (e.g., loss of a key developer).
3. Calculate the Schedule Variance (SV) and Cost Variance (CV) for a project using Earned Value Management (EVM) data.

Textbooks:

1. Bob Hughes, Mike Cotterell & Rajib Mall, Software Project Management, 5th Edition, McGraw Hill, ISBN: 9780077122799
2. Roger S. Pressman, Software Engineering: A Practitioner's Approach, 10th Edition, McGraw Hill, ISBN: 9789355326771

Reference Books:

1. Sommerville, Software Engineering, 9th Edition, Pearson Education Ltd, ISBN: 9780137035151
2. Pankaj Jalote, An Integrated Approach to Software Engineering, 3rd Edition, Narosa Publishing House, ISBN: 9788173192714

Web Links:

1. https://onlinecourses.nptel.ac.in/noc25_cs109/preview
2. https://se-iitkgp.vlabs.ac.in

AGILE METHODOLOGIES

Course Code: 2603MC21	**L**	**T**	**P**	**C**
	2	**0**	**2**	**4**

Course Outcomes:

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

CO1: Understand software lifecycle and need for agility,

CO2: Create the UML diagrams using smartDraw

CO3: Setup and Create Scrum board and Kanban board

CO4: Apply customer collaboration, bottleneck resolution, and Git for CI/CD.

CO5: Implement Scrum for team formation and scaling, track project metrics.

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	PO9	PO10
CO1	1	2	2	-	1	-	1	-	1	-
CO2	2	3	3	2	1	-	1		1	-
CO3	1	2	3	2	1	1	-	-	1	-
CO4	1	2	3	2	1	1	-	-	1	-
CO5	1	2	3	2	1	1	-	-	1	1

Mapping of Course Outcomes with Program Specific Outcomes:

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

UNIT – I

Agile Development: Agility, Agility and Cost of change, Agile Process, Agility Principles, Agile Process Models.

Practice:

1. Get an understanding of the stages in software development lifecycle, the process Models.
2. Get an understanding of the values and principles of agility and the need for agile software development.

UNIT – II

Unified Modeling Language: UML Diagrams, Class Diagram, Activity Diagram, Use Case Diagram

Practice:

1. Get an understanding of the Object Oriented Project Design
2. Creating UML diagrams using smartdraw(https://app.smartdraw.com/)

UNIT – III

Backdrop-The Science of Scrum: The Skelton and Heart of Scrum, Scrum Roles, Scrum Flow, Scrum Artifacts.

Practice:

1. Set up a Kanban board and demonstrate the workflow.
2. Create a Scrum board using tools like Jira or Trello.
3. Discuss and document the responsibilities of Scrum roles in different case scenarios.

UNIT – IV

The Product Owner and Introduction To Ci/Cd: Customer and team Collaboration, Introduction to Continuous Integration, Continuous Delivery and Deployment, Benefits of CI/CD, Metrics to track CICD practices.

Practice:

1. Configure the web application and Version control using Git using Git commands and version control operations.
2. Create a Git Staging Environment using Git Bash
3. Create and manage branches in a GitHub using commands in Command Prompt

UNIT – V

Sprint Planning Meeting, Daily Scrum Meeting, Sprint, Sprint Review Meeting, Sprint Retrospective meeting.

Practice:

1. Create and manage Scrum artifacts (Product Backlog, Sprint Backlog, Burndown Chart) for a project.
2. Simulate Sprint Planning, Daily Stand-ups, Sprint Review, and Retrospective meetings.

Additional Practice:

1. Git Staging Environment using Git Bash
2. Create a Clone for the Repository in GitHub
3. Diagnose your team with the Scrum Team Survey.

Textbooks:

1. Software Engineering A PRACTITIONER'S APPROACH, Roger S. Pressman, Ph.D. SEVENTH EDITION, McGraw-Hill
2. Agile Project Management with Scrum, Ken Schwaber, 1st Edition,Microsoft Press, ISBN: 978-0735619937.

Reference Books:

1. Agile Software Engineering, Series: Undergraduate Topics in Computer Science, Hazzaand Dubinsky, Springer, ISBN: 978-3319251319.
2. What is Devops? Infrastructure as code, 1st Edition, Mike Loukides ,O'Reilly Publications, ISBN: 978-1491924358.
3. Agile Software Development, Cockburn, Highsmith, Alistair Cockburn Series, Addison-Wesley, ISBN: 978-0321249166.

Web Links:

1. https://www.edx.org/course/agile-software-development
2. https://www.class-central.com/course/coursera-agile-software-development-9513
3. https://www.cprime.com/resources/what-is-agile-what-is-scrum/
4. https://www.atlassian.com/agile/kanban
5. https://file.scirp.org/pdf/JCC_2017033115471602.pdf

SOFTWARE TESTING MANAGEMENT

Course Code: 2603MC22

L	T	P	C
3	0	1	4

Course Outcomes:

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

CO1: Design rigorous test cases using Black-box and White-box methodologies.

CO2: Integrate manual testing results with Automated testing frameworks.

CO3: Manage the software maintenance lifecycle and regression testing.

CO4: Perform Reverse Engineering to reconstruct legacy system specifications

CO5: Evaluate software reliability and implement Business Process Re-engineering.

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	PO9	PO10
CO1	2	3	2	2	-	-	-	-	-	1
CO2	-	-	1	3	-	-	-	-	-	2
CO3	1	2	1	-	-	-	-	-	2	1
CO4	3	2	2	1	-	1	-	-	-	2
CO5	2	1	2	-	-	-	-	1	-	2

Mapping of Course Outcomes with Program Specific Outcomes:

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

UNIT – I

Testing Fundamentals and Design Strategies: The Psychology of Testing; Objectives and Principles of Software Testing; Exhaustive vs. Effective testing; The Testing Process: Planning, Execution, and Reporting; Test Case Design: Verification vs. Validation; Static Testing: Walkthroughs and Inspections; Dynamic Testing: Behavioral vs. Structural approaches.

Practice:

1. Design 5 test cases for a "User Registration Form" focusing on Validation testing.
2. Perform a simulated "Walkthrough" of a code snippet and document errors found.
3. Draft a "Bug Report" for a defect including severity, priority, and steps to reproduce.

UNIT – II

Automation and Formal Technical Reviews: Manual vs. Automated Testing: Criteria for Automation; Introduction to Automation Frameworks and Tools (Selenium, JUnit); Formal Technical Reviews (FTR): Review Meeting, Review Reporting, and Record-keeping; Review Guidelines; Testing in the DevOps Environment: Continuous Integration and Continuous Testing.

Practice:

1. Select 3 manual test cases and justify why they should or should not be automated.
2. Conduct an FTR on requirement specifications and prepare a formal Review Report.
3. Create a basic automated test script using Selenium or JUnit for a simple web login.

UNIT – III

Structural and Functional Testing Techniques: White Box Testing: Path Testing, Basis Path Testing, Control Structure Testing (Condition, Data Flow, Loop); Black Box Testing: Boundary Value Analysis (BVA), Equivalence Class Partitioning, Cause-Effect Graphing; Specialized Testing: Unit Testing, Integration Testing (Top-down, Bottom-up, Sandwich), System Testing, and User Acceptance Testing (UAT).

Practice:

1. Use BVA to design test cases for an input field accepting ages between 18 and 60.
2. Create a Basis Path Testing suite by calculating Cyclomatic Complexity for code.
3. Prepare an Integration Testing plan (Top-down) for a system of four dependent modules.

UNIT – IV

Software Maintenance and Regression Dynamics: Software Maintenance: Introduction and Need; Types of Maintenance: Corrective (Bug fixing), Adaptive (Environment changes), Perfective (Feature enhancement), and Preventive; The Software Project Maintenance Life Cycle (SPMLC); Regression Testing: Strategies and Selection of Test Cases; Impact Analysis

Practice:

1. Classify 5 maintenance requests into maintenance types.
2. Design a Regression Testing suite for software after a minor security patch.

UNIT – V

Reverse Engineering and System Modernization: Legacy Systems: Challenges and Risks; Concepts of Reverse Engineering: Abstraction Levels, Program Understanding; Business Process Re-engineering (BPR): Strategy and Implementation; Software Restructuring and Code Refactoring; Forward Engineering: Re-building from Re-engineered models; The Economics of Re-engineering.

Practice:

1. Analyze provided Legacy source code and recreate its Flow Chart and DFD.
2. Propose a Refactoring plan for code to improve modularity.

Textbooks:

1. Anirban Basu, Software Quality Assurance, Testing and Metrics, 1st Edition, PHI Learning, ISBN: 9788120339988
2. Srinivasan Desikan and Gopalaswamy Ramesh, Software Testing: Principles and Practices, Pearson Education, ISBN: 9788177581218
3. Aditya P. Mathur, Foundations of Software Testing, 2nd Edition, Pearson Education, ISBN: 9788131759080

Reference Books:

1. Ian Sommerville, Software Engineering, 9th Edition, Pearson Education Ltd, ISBN: 9780137035151
2. Pankaj Jalote, An Integrated Approach to Software Engineering, 3rd Edition, Narosa Publishing House, ISBN: 9788173192714

Web Links:

1. https://onlinecourses.nptel.ac.in/noc20_cs68/preview
2. https://se-iitkgp.vlabs.ac.in

ETHICAL HACKING

Course Code: 2603MC23

L	T	P	C
3	0	1	4

Course Outcomes:

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

CO1: Explain fundamental cyber security concepts, attack taxonomy, security models, risk management, and emerging cyber threats.

CO2: Describe hacking concepts, phases, networking basics, and apply footprinting and scanning techniques to identify security exposures

CO3: Analyze penetration testing processes, system hacking methods, password cracking techniques, and malware behavior

CO4: Examine network and human-centric attacks such as sniffing, session hijacking, phishing, and social engineering techniques.

CO5: Explain cryptographic principles, steganography methods, and vulnerability assessment frameworks such as OWASP for securing 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	PO9	PO10
CO1	3	2	1	1	-	-	-	-	-	-
CO2	-	3	2	1	2	-	-	-	-	-
CO3	-	1	3	2	2	-	-	-	-	-
CO4	-	-	-	3	2	-	-	-	-	-
CO5	1	-	-	-	3	-	-	-	2	1

Mapping of Course Outcomes with Program Specific Outcomes:

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

UNIT – I

Introduction to Cyber Security: Basic Cyber Security Concepts, layers of security, Vulnerability, threat, Harmful acts, Internet Governance – Challenges and Constraints, Computer Criminals, CIA Triad, Assets and Threat, motive of attackers, active attacks, passive attacks, Software attacks, hardware

attacks, Spectrum of attacks, Taxonomy of various attacks, IP spoofing, Methods of defense, Security Models, risk management, Cyber Threats-Cyber Warfare, Cyber Crime, Cyber terrorism

Practice:

1. Identifying Vulnerable Machines & Environments
2. Attack Classification & Defense Mapping – Case studies

UNIT – II

Introduction to Hacking: Hacking, Types and phases of hacking, Introduction to Ports & Protocols: Ports, Protocols, Primary Network Types. Footprinting: Footprinting, Types, Using ping and ns Lookup commands in Windows command line, Scanning: Scanning, Basics of Scanning, Basic Techniques of Scanning.

Practice:

1. Footprinting Using Command-Line Tools
2. Beginner Labs on Online Platforms (HTB , THM)

UNIT – III

Introduction to Penetration Testing: Penetration test, Categories and Types of Penetration tests, Structure of Penetration Test Report. System Hacking, Password Cracking, Default password databases, Manual and Automated Password Cracking, Process of System Hacking, Using Keyloggers, Trojans & Backdoors: Trojans, Working of Trojan, Infection Techniques, Attack, Classification of Virus.

Practice:

1. Penetration Testing Lifecycle Simulation.
2. Capture The Flag (CTF) Challenges

UNIT – IV

Sniffing, Packet Analysis, Types of Sniffing, Active and Passive Sniffing Techniques, Session Hijacking, Social Engineering: Social Engineering, Process, Phishing Process, Types of Phishing Attacks, Social Engineering Toolkit (SET)

Practice:

1. Packet Sniffing &Analysis
2. Social Engineering Analysis using SET

UNIT – V

Cryptography: Cryptography, Digital Signature, Hash Functions, Steganography: Steganography Process, watermarking, Steganography Methods and Attacks, Steganography tools, Vulnerability Assessment: Vulnerability, The Open Web Application Security Project

Practice:

1. Steganography Techniques & Detection
2. Vulnerability Assessment Using OWASP

Textbooks:

1. Harsh Bothra, Hacking: Be a Hacker with Ethics, Khanna Publications Ist Edition ISBN: 9386173050.
2. Prakhar Prasad, Mastering Modern Web Penetration Testing, Packt Publishing ISBN-13: 978-1785284588

Reference Books:

1. Rafay Baloch, Ethical Hacking and Penetration Testing Guide, ISBN: 9781138436824
2. Christopher Hadnagy Social Engineering: The Science of Human Hacking" by John Wiley & Sons , 9780470639535 (ISBN10: 0470639539)

Web Links:

1. www.geeksforgeeks.org/introduction-to-ethical-hacking
2. https://www.coursera.org/learn/ethical-hacking-basics?action=enroll#modules
3. https://nptel.ac.in/courses/106105217

BLOCK CHAIN TECHNOLOGIES

Course Code: 2603MC24

L	T	P	C
3	0	1	4

Course Outcomes:

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

CO1: Demonstrate the foundation of the Block Chain technology and understand the processes in payment and funding.

CO2: Choose the present landscape of Block chain implementations and understand Crypto currency markets.

CO3: Review of legal implications using smart contracts.

CO4: Understanding the Ethereum Virtual Machine (EVM):

CO5: Analyze security, privacy, and efficiency of a given block chain 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	PO9	PO10
CO1	3	2	2	2	1	1	-	-	1	2
CO2	1	3	2	-	-	-	2	-	-	1
CO3	1	1	-	-	2	2	3	2	1	1
CO4	3	-	2	1	-	-	1	2	1	-
CO5	2	-	-	1	1	1	3	-	-	2

Mapping of Course Outcomes with Program Specific Outcomes:

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

UNIT – I

Introduction: What is Blockchain (BC), public ledgers, BC as public ledgers; BC history - Bitcoin and Cryptocurrency, BC 2.0, Smart contracts; BC architecture – Blocks in BC, transactions and distributed consensus.

Practice:

1. Simulate a simple blockchain using Python
2. Create blocks containing transaction data, previous hash, timestamp

UNIT – II

Cryptocurrency: Cryptographic basics for cryptocurrency, A short overview of Hashing, signature schemes, Public Key Cryptography.

Practice:

1. Implement hashing (SHA-256) for message integrity.
2. Sign and verify a message using digital signatures.

UNIT – III

Bitcoin: Bitcoin Wallet, Blocks, Merkley Tree, hardness of mining, transaction verifiability, anonymity, forks, double spending, mathematical analysis of properties of Bitcoin. Blockchain Consensus Algorithms – Proof-of-Work (POW).

Practice:

1. Implement Proof-of-Work by finding a nonce satisfying difficulty conditions.
2. Verify a transaction using Merkle proof

UNIT – IV

Ethereum: Ethereum Virtual Machine (EVM), Wallets for Ethereum, Solidity, Smart Contracts, some attacks on smart contracts. Scaling of Blockchains.

Practice:

1. Compile and deploy on local Ethereum network

UNIT – V

Trends and Topics: Zero Knowledge proofs and protocols in Blockchain, Succinct non- interactive argument for Knowledge (SNARK), pairing on Elliptic curves.

Practice:

1. Simulate a simple Zero-Knowledge Proof (ZKP) example.
2. Demonstrate proof without revealing secret information.

Textbooks:

1. Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction", Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller, and Steven Gold feder,1st Edition, Princeton University Press, ISBN: 978-0691171692.
2. Blockchain: Blue print for a New Economy, Melanie Swan, O'Reilly, ISBN:1491920475,9781491920473.

Reference Books:

1. Research Perspectives and Challenges for Bitcoin and Cryptocurrency", Joseph Bonneau, Andrew Miller, Jeremy Clark, Arvind Narayanan, Joshua A. Kroll, Edward W. Felten, IEEE Symposium on Security and Privacy, ISBN: 978-1-4673-6949-7.
2. The Bitcoin Backbone Protocol: Analysis and Applications, Juan A. Garay, Aggelos Kiayias, Nikos Leonardos, EUROCRYPT 2015, Part II, Lecture Notes in Computer Science (LNCS), Volume 9057, Springer, ISBN: 978-3662469607
3. Mastering Ethereum: Building Smart Contracts and D App's by Andrews ISBN-13. 978-1491971949.

Web Links:

1. https://builtin.com/blockchain
2. https://www.computerworld.com/article/3191077/what-is-blockchain-the-complete guide.html
3. https://blockgeeks.com/guides/what-is-blockchain-technology/F

FUNDAMENTALS OF DATA SCIENCE

Course Code: 2603MC25	**L**	**T**	**P**	**C**
	2	**0**	**2**	**4**

Course Outcomes:

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

CO1: Explain core concepts, terminology, and the data science lifecycle.

CO2: Collect, clean, explore, and summarize data using basic tools.

CO3: Apply critical thinking to interpret data and avoid misleading conclusions.

CO4: Communicate data insights effectively using visual and written formats.

CO5: Examine ethical, societal, and environmental issues related to data 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	PO9	PO10
CO1	3	1	-	1	-	-	2	-	-	--
CO2	3	1	-	-	2	-	2	-	-	
CO3	3	2	1	-	2	-	-	-	-	-
CO4	2	3	2	-	-	-	-	-	-	-
CO5	1	1	-	3	2	2	-	-	-	-

Mapping of Course Outcomes with Program Specific Outcomes:

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

UNIT – I

Introduction to Data Science: Definition and scope of Data Science, Data vs Information vs Knowledge, Data Science lifecycle, Types of data: structured, semi-structured, unstructured, Applications of data science in business, governance, healthcare, education, Role of data science in decision-making

Practice:

1. Identify a real-life system (college, hospital, bank, mobile application, traffic system) and list the types of data generated in it.
2. Classify the identified data into structured, semi-structured, and unstructured categories.
3. Explain how the collected data can support decision-making in the chosen system.

UNIT – II

Data Collection and Data Preparation: Sources of data: surveys, web, sensors, open data portals, Introduction to datasets (CSV, Excel, simple formats), Data quality issues: missing data, noise, inconsistency, Data cleaning and pre-processing (conceptual + simple practical examples), Introduction to data privacy and informed consent

Practice:

1. In a given small dataset as CSV or Excel format, identify the attributes and records present in the dataset.
2. Examine the dataset and list any missing values, inconsistent entries, or incorrect records.
3. Perform basic data cleaning steps such as removing duplicates or correcting inconsistent values.
4. Does the dataset contain personal or sensitive information? Discuss whether informed consent would be required for its use.

UNIT – III

Exploratory Data Analysis and Descriptive Statistics: Introduction to Exploratory Data Analysis (EDA), Measures of central tendency: mean, median, mode, Measures of dispersion: range, variance, standard deviation, Data summarization techniques, interpreting results and avoiding misleading conclusions

Practice:

1. For a given dataset, calculate the mean, median, and standard deviation of a selected attribute.
2. Compare the measures of central tendency and comment on which value best represents the data.
3. Identify any unusual or extreme values in the dataset and discuss their possible impact on analysis.

UNIT – IV

Data Visualization and Communication: Importance of data visualization, Types of charts: bar chart, line chart, pie chart, histogram, Choosing the right visualization, Basics of storytelling with data, Common visualization errors and biases

Practice:

1. Create a bar chart or line chart to represent a selected attribute from the dataset.
2. Interpret the visualization and state the key pattern or trend observed.
3. Identify one possible misleading interpretation that could arise from the visualization and explain why.

UNIT – V

Ethics, Society, Sustainability and Mini Project:

Ethics & Social Impact: Ethics in data science, Data bias and fairness, Social implications of data-driven decisions, Environmental impact of data storage and computing

Mini Project (Practical Component): Problem identification, dataset selection, basic data analysis and visualization, interpretation of results, ethical reflection, and group presentation.

Practice:

1. Select a simple real-world problem that can be analyzed using data (attendance, pollution levels, sales trends, survey responses).
2. Identify an appropriate dataset and perform basic analysis and visualization.
3. Discuss any ethical, social, or environmental concerns related to the data or the conclusions drawn.
4. Present the findings in a brief group report or presentation.

Additional Practice:

1. Data Exploration Practice: Students shall explore a small real-world dataset to identify data types, missing values, and basic summary statistics (mean, median, standard deviation) using spreadsheet tools or simple software.
2. Visualization and Interpretation: Students shall create basic visualizations (bar chart, line chart, histogram, box plot) and interpret the results, identifying patterns, trends, and possible biases.
3. Statistical Reasoning Exercise: Students shall formulate simple null and alternative hypotheses for given scenarios and identify appropriate statistical tests (Z-test, T-test, Chi-square, correlation) at a conceptual level.
4. Mini Case Study / Group Activity: Students shall work in small groups on a case study involving data-driven decision-making, addressing ethical, social, or sustainability concerns, and present their findings.

Textbooks:

1. Moolayil, Jojo, Smarter Decisions: The Intersection of Internet of Things and Decision Science, First Edition, Packt Publishing, 2016. ISBN: 978-1-78588-419-1.
2. O'Neil, Cathy and Schutt, Rachel, Doing Data Science: Straight Talk from the Frontline, First Edition, O'Reilly Media, Inc., 2013. ISBN: 978-1-4493-5865-5.

Reference Books:

1. Dietrich, David, Heller, Barry and Yang, Beibei, Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data, First Edition, Wiley-India, 2015. ISBN: 978-1-118-87613-8.
2. Raj, Pethuru, Handbook of Research on Cloud Infrastructures for Big Data Analytics, IGI Global, 2014. ISBN: 978-1-4666-5864-6.

Web Links:

1. https://github.com/a00ayad00/Machine-Learning-and-Data-Science-Resourses
2. https://www.mltut.com/free-courses-on-coursera-for-data-science/
3. https://www.udacity.com/course/data-scientist-nanodegree--nd025
4. https://huggingface.co/papers/trending

DATA ANALYTICS

Course Code: 2603MC26

L	T	P	C
3	0	1	4

Course Outcomes:

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

CO1: Understand basic concepts, importance, and applications of Data Analytics.

CO2: Identify different types of data and perform basic data preprocessing.

CO3: Apply simple statistical measures to analyze data.

CO4: Use basic data visualization techniques to interpret data.

CO5: Use basic data visualization techniques to interpret 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	PO9	PO10
CO1	3	2	-	-	-	-	-	1	-	-
CO2	3	2	1	-	-	-	-	-	-	-
CO3	3	2	1	-	-	-	-	-	-	-
CO4	2	2	1	1	1	-	-	1	-	-
CO5	3	2	1	1	-	-	-	-	-	-

Mapping of Course Outcomes with Program Specific Outcomes:

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

UNIT – I

Introduction to Data Analytics: Definition of data analytics, importance of data analytics, data-driven decision making, applications of data analytics in business, healthcare, education, and government. **Types of data:** structured, semi-structured, and unstructured data. Challenges in data analytics.

Practice:

1. Define data analytics and explain its importance.
2. List applications of data analytics.
3. Differentiate between structured and unstructured data.

UNIT – II

Data Collection and Preprocessing: Sources of data and methods of data collection. Data quality issues. **Data preprocessing techniques**: data cleaning, handling missing values, data transformation, and data reduction

Practice:

1. Explain different sources of data.
2. What is data preprocessing? Why is it required?
3. Explain data cleaning with an example.

UNIT – III

Basic Statistical Analysis: Introduction to statistics for data analytics. **Measures of central tendency:** mean, median, and mode. **Measures of dispersion:** range and standard deviation. Introduction to correlation.

Practice:

1. Explain mean, median, and mode.
2. What is standard deviation?

UNIT – IV

Data Visualization: Need for data visualization. **Types of charts and graphs:** bar chart, line chart, pie chart, histogram, and scatter plot. Principles of effective data visualization.

Practice:

1. Explain the importance of data visualization.
2. List different charts used in data analytics.

UNIT – V

Data Analytics Tools and Applications: Introduction to basic data analytics tools (spreadsheets such as MS Excel / Google Sheets). Basic data analysis operations using spreadsheets. Simple real-world case studies. Ethical issues and data privacy in data analytics

Practice:

1. Explain the role of spreadsheets in data analytics.
2. Discuss real-world applications of data analytics.
3. Write short notes on ethical issues in data analytics.

Textbooks:

1. Anil Maheshwari, Data Analytics Made Accessible, First Edition, McGraw-Hill Education, ISBN: 978-1259028144
2. Seema Acharya and Subhashini Chellappan, Big Data and Analytics, First Edition, Wiley India, ISBN: 978-8126554780

Reference Books:

1. Vijay Kotu and Bala Deshpande, Predictive Analytics and Data Mining, First Edition, Morgan Kaufmann, ISBN: 978-0128014608
2. Wes McKinney, Python for Data Analysis (selected concepts), Second Edition, O'Reilly Media, ISBN: 978-1491957660

Web Links:

1. https://www.tutorialspoint.com/data_analytics/
2. https://www.geeksforgeeks.org/data-analytics/
3. https://nptel.ac.in/courses/

EXPERT SYSTEMS

Course Code: 2603MC27

L	T	P	C
3	0	1	4

Course Outcomes:

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

CO1: Understand the fundamental components of an expert system

CO2: Identify suitable problem domains for expert system development

CO3: Understand the working principles of different types of expert systems,

CO4: Understand the fundamentals of probability theory and its role in reasoning under uncertainty in intelligent systems.

CO5: Identify and apply different measures for matching 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	PO9	PO10
CO1	2	1	2	-	-	-	-	-	-	-
CO2	2	1	2	1	-	-	-	-	2	-
CO3	1	1	2	1	-	-	-	-	1	-
CO4	1	2	1	2	-	-	-	-	-	-
CO5	1	2	1	2	-	-	-	-	-	-

Mapping of Course Outcomes with Program Specific Outcomes:

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

UNIT – I

Expert System: Components of Expert System: Knowledge Base, Inference Engine, User Interface, Features of Expert System, Expert System Life Cycle, Categories of Expert System, Rule Based vs. Model Based Expert Systems, Advantages/Limitations of Expert System.

Practice:

1. Introduction to LISP: Syntax and Numeric Function
2. Basic List Manipulation Functions in LISP
3. Functions, Predicates, and Conditionals

UNIT – II

Developing an Expert System: Identification, Conceptualization, Formalization, Implementation, Testing, Using an Expert System, Application Areas of Expert System, phases in building expert systems, expert system architecture, expert system versus traditional systems, rule-based expert systems.

Practice:

1. Input, Output, and Local Variable
2. Iteration and Recursion
3. property Lists and Array

UNIT – III

Frame-based expert systems, Fuzzy logic based expert systems, Neural network based expert systems, Natural Language interface for Expert Systems, Blackboard Expert System – HEARSAY

Practice:

1. Vectors in Lisp
2. Set in Lisp
3. Set using List vs Set using Hash table

UNIT – IV

Probability theory: Introduction, probability theory, Bayesian belief networks, certainty factor theory, dempster- shafer theory

Practice:

1. Inorder, preorder and Postorder Traversal of Tree

UNIT – V

MATCHING TECHNIQUES: Structures Used in Matching, Measures for Matching, Matching Like Patterns, Partial Matching, Fuzzy Matching Algorithms, The RETE Matching Algorithm.

Practice:

1. Depth First Traversal of Tree

Textbooks:

1. Introduction to ARTIFICIAL INTELLIGENCE AND EXPERT SYSTEMS, DAN. W. PATTERSON, prentice hall of india private limited, New Delhi, 2002
2. Principles of Expert Systems, Peter J.F. Lucas & Linda C. van der Gaag Centre for Mathematics and Computer Science, Amsterdam, published in 1991 by Addison-Wesley

Reference Books:

1. Foundation of Artificial Intelligence and Expert Systems by V.S. Janakiraman, K. Sarukesi, P. Gopalakrishnan, Mc Millan
2. Expert Systems Principles and Programming (3rd Edition) by Giarratano & Riley, Thomson (Vikas Publishing House)

Web Links:

1. https://www.tutorialspoint.com/lisp/index.htm
2. https://www.geeksforgeeks.org/lisp/introduction-to-lisp/
3. https://aticleworld.com/lisp-programming-tutorial/

DEEP LEARNING

Course Code: 2603MC28

L	T	P	C
3	0	1	4

Course Outcomes:

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

CO1: Understand the basics of neural networks and their use in deep learning
CO2: Implement and use convolutional neural networks for image analysis
CO3: Implement recurrent neural networks for sequential data tasks
CO4: Apply autoencoders, VAEs, and GANs for unsupervised learning
CO5: Understand restricted Boltzmann machines and their 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	PO9	PO10
CO1	3	2	-	1	-	-	-	3	-	2
CO2	2	1	2	3	-	-	-	-	-	2
CO3	2	3	-	1	-	-	-	1	-	2
CO4	2	2	-	3	-	1	-	-	-	2
CO5	2	2	-	3	-	-	-	-	-	2

Mapping of Course Outcomes with Program Specific Outcomes:

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

UNIT – I

Introduction to Deep Learning: Artificial Intelligence, Machine Learning, Deep Learning, Neural Networks (Biological Inspiration, Perceptron Model), Structure of Neural Network (Input Layer, Hidden Layer, Output Layer), Training of Neural Networks (Forward Propagation, Backpropagation – conceptual), Activation Functions (Sigmoid, Tanh, ReLU), Loss Functions (Mean Squared Error, Cross Entropy Loss), Hyperparameters (Learning Rate, Epochs, Batch Size), Autoencoders (Basic Concept)

Practice:

1. Explain the basic structure of an Artificial Neural Network (ANN) with a neat diagram. Describe the role of input, hidden, and output layers.
2. Using a real-world example (such as house price prediction or handwritten digit recognition), explain how data flows through each layer. Also illustrate the concept using a simple Python/Keras model.

UNIT – II

Convolutional Neural Networks (CNN): Convolutional Neural Networks, Image Representation, CNN Architecture (Convolution Layer, Pooling Layer, Fully Connected Layer), Pooling Techniques (Max Pooling), Activation Functions in CNN (ReLU, ELU), Advantages of CNN, CNN Architectures (LeNet, AlexNet – overview), Applications of CNN (Image Classification, Face Recognition, Handwritten Digit Recognition)

Practice:

1. What is a Convolutional Neural Network (CNN)? Explain the architecture of a CNN with a neat diagram, describing the role of convolution layers, pooling layers, activation functions, fully connected layers, and output layers.
2. Explain the use of CNNs for handwritten digit recognition using the MNIST dataset.

UNIT – III

Recurrent Neural Networks (RNN): Sequential Data, Recurrent Neural Networks, RNN Architecture (Hidden State, Feedback Connections), Training Challenges (Vanishing Gradient Problem – conceptual), Long Short-Term Memory (LSTM, Gates – overview), Applications of RNN (Text Processing, Time Series Prediction)

Practice:

1. Design and implement a basic RNN model for processing sequential data. Describe its architecture and observe its limitations during training.
2. Implement an LSTM network for the same task and experimentally demonstrate how it overcomes the limitations of a simple RNN.

UNIT – IV

Unsupervised and Generative Learning: Unsupervised Learning, Autoencoders (Basic Autoencoder, Sparse Autoencoder – overview, Denoising Autoencoder), Variational Autoencoders (High-Level Overview), Generative Adversarial Networks (Generator, Discriminator, Training Process), Applications of Generative Models

Practice:

1. Explain Autoencoders and their types. Discuss the basic working principle of Generative Adversarial Networks (GANs).

UNIT – V

Deep Learning Models and Applications: Energy-Based Models, Restricted Boltzmann Machines (Structure, Sampling – basic idea), Deep Belief Networks (Overview), Applications of Deep Learning (Handwritten Character Recognition, Face Recognition, Social Networks, Recommendation Systems)

Textbooks:

1. Deep Learning - Ian Goodfellow, Yoshua Bengio, Aaron Courville, 1st Edition, The MIT Press publication, ISBN-10: 0262035618, ISBN-13: 978-0262035613
2. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems - Aurélien Géron, Edition 3, O'Reilly Media Publication on 2022-11-08, ISBN-13: 9781098125974, ISBN-10: 1098125975

Reference Books:

1. Dive into Deep Learning - Aston Zhang, Zachary C. Lipton, Mu Li, Alexander J. Smola, 1st Edition, Cambridge University Press on 2023, ISBN-13: 9781009389433
2. Deep Learning: Foundations and Concepts - K. Taylor, 1st Edition, Publisher: Springer, Publication Date: 2023, ISBN-13: 9783031454677, ISBN-10: 3031454677

Web Links:

1. http://deeplearning.stanford.edu/tutorial/
2. https://nptel.ac.in/courses/106/106/106106184/
3. https://www.coursera.org/specializations/deep-learning
4. https://www.geeksforgeeks.org/introduction-deep-learning/

INFORMATION RETRIEVAL SYSTEMS

Course Code: 2603MC29

L	T	P	C
3	1	0	4

Course Outcomes:

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

CO1: Ability to apply Pre-processing for effective information retrieval.

CO2: Ability to Apply indexing in information retrieval.

CO3: Understand the index compression process.

CO4: Transform textual information into vectors.

CO5: Analyze ranked and unranked search results.

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	PO9	PO10
CO1	2	1	2	1	1	-	-	-	-	-
CO2	2	2	1	1	-	-	1	-	-	-
CO3	2	1	2	1	-	-	-	-	-	-
CO4	2	1	2	1	-	-	-	-	1	-
CO5	2	1	2	1	-	-	-	-	1	-

Mapping of Course Outcomes with Program Specific Outcomes:

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

UNIT – I

Boolean Retrieval: An example information retrieval problem, A first take at building an inverted index, Processing Boolean queries, The extended Boolean model versus ranked retrieval. The Term vocabulary and postings lists : Document delineation and character sequence decoding, Obtaining the character sequence in a document, Choosing a document unit, Determining the vocabulary of terms ,Tokenization, Dropping common terms: stop words, Normalization (equivalence classing of terms) stemming and lemmatization, Faster postings list intersection via skip pointers, Positional postings and phrase queries , Biword indexes , Positional indexes , Combination schemes

UNIT – II

Dictionaries and tolerant retrieval: Search structures for dictionaries, Wildcard queries, General wildcard queries, k-gram indexes for wildcard queries, Spelling correction Implementing spelling correction, Forms of spelling correction, Edit distance, k-gram indexes for spelling correction, Context sensitive spelling correction, Phonetic correction.

Index construction: Hardware basics, Blocked sort-based indexing, Single-pass in-memory indexing, Distributed indexing, Dynamic indexing, other types of indexes.

UNIT – III

Index compression: Statistical properties of terms in information retrieval, Heaps' law: Estimating the number of terms, Zipf's law: Modeling the distribution of terms, Dictionary compression, Dictionary as a string, Blocked storage, Postings file compression, Variable byte codes, ã codes.

Scoring, term weighting: Parametric and zone indexes, Weighted zone scoring, Learning weights, The optimal weight g, Term frequency and weighting, Inverse document frequency, Tf-idf weighting.

UNIT – IV

The vector space model: The vector space model for scoring, Dot products, Queries as vectors, Computing vector scores, Variant tf-idf functions, Sublinear tf scaling, Maximum tf normalization, Document and query weighting schemes, Pivoted normalized document length.

UNIT – V

Evaluation in information retrieval: Information retrieval system evaluation, Standard test collections, Evaluation of unranked retrieval sets, Evaluation of ranked retrieval results, assessing relevance, Critiques and justifications of the concept of Relevance, A broader perspective: System quality and user utility, System issues, User utility, Refining a deployed system, Results snippets.

Textbooks:

1. Christopher D. Manning, Prabhakar Raghavan, Hinrich Schütze, An Introduction to Information Retrieval, 1st Edition, Cambridge University Press, 2008.
2. "Information Storage and Retrieval Systems" – Theory and Implementation, 2nd Edition, Gerald J. Kowalski, Mark T. Maybury, Springer. ISBN-13: 978- 0387406038.

Reference Books:

1. G.G. Chowdhury, Introduction to Modern Information Retrieval, 3 rd Edition, neal-schuman publishers, 2010.
2. Information Retrieval Data Structures and Algorithms, 1st Edition, Prentice Hall. Frakes, W.B., Ricardo Baeza-Yates: ISBN-13: 978-0131661999
3. Information Storage & Retrieval , 1st Edition, By Robert Korfhage – John Wiley & Sons. ISBN-13: 978-0471048688.

Web Links:

1. https://www-nlp.stanford.edu/IR-book/
2. https://www.geeksforgeeks.org/what-is-information-retrieval/
3. https://www.tutorialspoint.com/natural_language_processing/natural_language_processing_information_retrieval.htm

FOUNDATIONS OF DATA ENGINEERING

Course Code: 2603MC30

L	T	P	C
3	0	1	4

Course Outcomes:

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

CO1: Describe the fundamental concepts of data engineering, data lifecycle, and the role of data engineers in modern data ecosystems

CO2: Apply techniques to ingest structured and unstructured data from various sources using batch and streaming methods.

CO3: Design and implement data storage solutions using RDBMS, NoSQL, data lakes, and modern file formats.

CO4: Develop data pipelines using tools such as Apache Spark, Kafka, and Airflow for batch and real-time processing.

CO5: Evaluate the performance, scalability, and quality of data workflows and pipelines based on modern data engineering 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	PO9	PO10
CO1	3	2	2	2	2	2	-	2	-	-
CO2	2	3	3	3	3	-	2	-	-	2
CO3	2	3	3	3	3	-	-	2	-	3
CO4	2	3	3	2	3	2	-	-	2	3
CO5	2	3	3	3	3	-	2	-	2	3

Mapping of Course Outcomes with Program Specific Outcomes:

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

UNIT – I

Introduction to Data Engineering: What is Data Engineering, The Data Lifecycle, Data Engineering vs. Data Science, Types of Data (Structured, Semi-structured, Unstructured), Roles and Responsibilities of a Data Engineer, Introduction to ETL (Extract, Transform, Load).

Practice:

1. Identify structured, semi-structured, and unstructured datasets (e.g., CSV, JSON, images).
2. Write a short report comparing Data Engineering vs. Data Science roles.

UNIT – II

Data Collection and Ingestion: Data Sources: APIs, Logs, Files, Databases, Data Ingestion Techniques: Batch vs. Stream, Tools: Apache Kafka, Apache Flume, Sqoop, Real-Time vs. Near-Real-Time Data Ingestion.

Practice:

1. Ingest log files into a database using batch processing.
2. Use Apache Kafka to stream sample messages and consume them in Python.
3. Compare batch vs. stream ingestion by loading the same dataset both ways.

UNIT – III

Data Storage and Management: Relational Databases (RDBMS): MySQL, NoSQL Databases: MongoDB, Data Warehousing: Snowflake, Google BigQuery, File Formats: CSV.

Practice:

1. Create tables in MySQL and insert sample records.
2. Store JSON documents in MongoDB and query them.
3. Load a dataset into Google BigQuery or Snowflake and run basic SQL queries.

UNIT – IV

Data Processing and Transformation: Batch Processing: Apache Spark, Stream Processing: Apache Flink, Data Cleaning and Validation, Data Transformation with Spark, Pandas, Workflow Orchestration: Apache Airflow.

Practice:

1. Use Apache Spark to perform word count on a text dataset.
2. Stream process live data (e.g., Twitter API or simulated logs) using Apache Flink.
3. Clean a dataset with Pandas (handle missing values, normalize columns).

UNIT – V

Data Pipelines and Deployment: Designing Scalable Data Pipelines, Monitoring and Logging, CI/CD for Data Pipelines.

Practice:

1. Design a pipeline that ingests, stores, and processes data end-to-end.
2. Implement logging and monitoring for a pipeline using Python logging or Spark UI.
3. Set up a CI/CD workflow for a data pipeline using GitHub Actions.

Textbooks:

1. "Designing Data-Intensive Applications" - Martin Kleppmann, March 2017, Publisher(s): O'Reilly Media, Inc., ISBN-13: 9781491903100.
2. "Fundamentals of Data Engineering" - Joe Reis and Matt Housley, June 2022, Publisher(s): O'Reilly Media, Inc., ISBN-13: 9781098108304.

Reference Books:

1. "Streaming Systems" - Tyler Akidau, Slava Chernyak, and Reuven Lax, July 2018. Publisher(s): O'Reilly Media, Inc. ISBN-13: 9781491983874.
2. "Big Data: Principles and Best Practices of Scalable Real-Time Data Systems" - Nathan Marz, Publisher: Dreamtech Press (1 January 2015), ISBN-13: 978-9351198062.

Web Links:

1. https://cloud.google.com/learn/learning-paths/data-engineer
2. https://github.com/DataTalksClub/data-engineering-zoomcamp
3. https://aws.amazon.com/blogs/big-data/

CLOUD COMPUTING

Course Code: 2603MC31

L	T	P	C
3	0	1	4

Course Outcomes:

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

CO1: Explore the various service and deployment models in cloud computing.

CO2: Demonstrate the basic concepts of virtualization and implementation levels of Virtualization.

CO3: Illustrate the architecture of cloud computing.

CO4: Apply Cloud programming and software environments on any real cloud service.

CO5: Analyse the Cloud Security risks and 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	PO9	PO10
CO1	3	2	-	1	-	-	-	1	-	2
CO2	2	3	2	1	-	-	-	-	-	1
CO3	2	2	3	1	-	-	-	-	-	1
CO4	2	1	2	3	2	1	-	1	2	2
CO5	1	2	1	2	-	1	3	2	-	2

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 Cloud Computing: Cloud models, Evolution of Cloud Computing, System models for Distributed and Cloud Computing, Cloud Computing Properties and Characteristics, Business Drivers for Adopting Cloud Computing, Software environments for distributed systems and clouds, Performance, Security and Energy Efficiency.

Practice:

1. Introduction to Cloud Computing.
2. Overview of Cloud Computing Concepts.
3. Hands-On: Setting up a free-tier account on AWS/AZURE/GCP.

UNIT – II

Virtual Machines and Virtualization of Clusters: Implementation Levels of Virtualization, Virtualization Structures/ Tools and mechanisms, Virtualization of CPU, Memory and I/O Devices, Virtual Clusters and Resource Management, Examples like VMware and Microsoft Hyper-V.

Practice:

1. Virtual Machines in Computer Services.
    a. Creating and Managing Virtual Machines.
    b. Hands-On: Launching and Configuring EC2 Instances on AWS.

UNIT – III

Cloud Platform Architecture: Cloud Computing and service Models, Architectural Design of Compute and Storage Clouds, Public Cloud Platforms, Service Oriented Architecture.

UNIT – IV

Programming Model: Parallel and Distributed Programming Paradigms, Map Reduce, Programming Support.

Security in The Cloud: Security Overview, Cloud Security Challenges, Access control mechanisms, Security Monitoring.

UNIT – V

Cloud Resource Management and Scheduling: Feedback Control Based on Dynamic Thresholds. Coordination of Specialized Autonomic Performance Managers, Resource Bundling, Scheduling Algorithms for Computing Clouds, Fair Queuing, Start Time Fair Queuing, Cloud Scheduling Subject to Deadlines.

Textbooks:

1. " Cloud Computing: Principles, Systems and Applications- Nick Antonopoulos, Lee Gillam; Springer Publication; ISBN: 978-1-84882-889-1.
2. Cloud Computing: Concepts, Technology & Architecture - Thomas Erl, Zaigham Mahmood, Ricardo Puttini; Prentice Hall Publication; ISBN: 978-0-13-338752-0

Reference Books:

1. Distributed and Cloud Computing: From Parallel Processing to the Internet of Things - Kai Hwang, Geoffrey C. Fox, Jack G. Dongarra; 1st Edition; Morgan Kaufmann Publishers; ISBN: 978-0-12-385880-8
2. Cloud Computing: Theory and Practice - Dan C. Marinescu; 1st Edition; Publication- Elsevier; ISBN: 978-0-12-404627-4

Web Links:

1. http://nptel.ac.in/courses/106106129/28
2. https://aws.amazon.com/getting-started/tutorials
3. https://www.coursera.org/learn/cloud-computing/lecture/VOIHP/introductionto-cloud-computing-concepts-part-1
4. https://www.lynda.com/Cloud-Computing-training-tutorials/1385-0.html

MOBILE APPLICATION DEVELOPMENT

Course Code: 2603MC32

L	T	P	C
2	0	2	4

Course Outcomes:

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

CO1: Understand fundamentals of mobile computing, Android architecture, and development environment.

CO2: Develop Android applications using Kotlin/Java and Android Studio.

CO3: Design interactive and user-friendly interfaces using layouts, views, and event handling.

CO4: Implement local and remote data storage using Shared Preferences, SQLite/Room database, and Web APIs.

CO5: Build, test, debug, and deploy complete Android applications through hands-on projects.

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	PO9	PO10
CO1	3	2	1	-	-	-	-	-	-	-
CO2	2	3	2	1	-	-	-	-	-	--
CO3	1	2	3	2	-	-	-	-	-	-
CO4	2	2	2	3	1	-	-	-	-	-
CO5	1	1	2	2	2	-	-	-	2	3

Mapping of Course Outcomes with Program Specific Outcomes:

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

UNIT – I

Introduction to Android & Setup: Basics of Mobile Computing, Android OS Architecture, Android Studio, SDK installation, Project structure, and Gradle basics

Practice:

1. Write a program to install Android Studio, configure the SDK, and create a Hello World application.
2. Implement an application to demonstrate Android project structure and resource management (strings, colours, layouts).

UNIT – II

Application Fundamentals: Activities and Activity lifecycle, Intents and navigation, Permissions, Services & Broadcast Receivers (overview)

Practice:

1. Write a program to demonstrate the Activity lifecycle using Logcat messages.
2. Implement an application with two or more screens using Intents and pass data between activities.

UNIT – III

User Interface Design: Views: TextView, Button, EditText, ImageView, Layouts: Linear, Constraint, RecyclerView, Event handling, Menus, Dialogs, Toast, Snackbar, Material Design basics

Practice:

1. Design and implement a form-based application using TextView, EditText, Button, and validation logic.
2. Create an application that displays a list of items using RecyclerView and custom layouts.

UNIT – IV

Data Storage & Networking: Shared Preferences, SQLite / Room database, JSON basics, REST APIs & Internet permissions, Firebase introduction

Practice:

1. Implement an application to store and retrieve user settings using Shared Preferences.
2. Create a CRUD application using SQLite/Room database to manage student or product records.

UNIT – V

Testing & Deployment: Debugging and Logcat, Testing basics, APK/AAB generation, Play Store publishing process, App performance & best practices

Practice:

1. Write a program to debug an application using Logcat and breakpoints.
2. Implement Firebase authentication (Login/Register) in an Android application.

Additional Practice:

1. Mini Project: Design and develop a project (To-Do app) and generate a signed APK/AAB for deployment.
2. Mini Project: Design and develop a project (Attendance app) and generate a signed APK/AAB for deployment.

Textbooks:

1. Horton, J., Android Programming for Beginners, 3rd Edition, Packt Publishing, 2021, ISBN: 978-1-800-56616-3.
2. Iversen, J., and Eierman, M., Learning Mobile App Development: A Hands-on Guide to Building Android and iOS Apps, 1st Edition, Addison-Wesley Professional, 2013, ISBN: 978-0-321-94786-4.

Reference Books:

1. Phillips, B., Stewart, C., Marsicano, K., and Gardner, B., Android Programming: The Big Nerd Ranch Guide, 4th Edition, Big Nerd Ranch Guides, 2019, ISBN: 978-0-134-70605-4.
2. Griffith, C., Mobile App Development with Ionic, Revised Edition, O'Reilly Media, 2017, ISBN: 978-1-491-99812-0.

Web Links:

1. https://onlinecourses.swayam2.ac.in/nou21_ge41/preview
2. https://nptel.ac.in/courses/106106147

NOSQL DATABASES

Course Code: 2603MC33

L	T	P	C
2	0	2	4

Course Outcomes:

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

CO1: Understand the fundamentals of NoSQL databases, their types, and their advantages over traditional relational databases.

CO2: Apply CRUD operations in NoSQL databases and work with document-oriented databases like MongoDB.

CO3: Implement key-value store concepts using Redis and Memcached for various use cases

CO4: Work with NoSQL query languages, indexing, and schema evolution in NoSQL databases.

CO5: Design, index, and optimize NoSQL databases for efficient data retrieval and storage.

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	PO9	PO10
CO1	3	2	1	-	-	-	-	-	-	1
CO2	2	1	3	-	-	1	-	-	-	-
CO3	1	3	-	2	-	-	-	1	-	-
CO4	-	1	2	-	3	-	1	-	-	-
CO5	1	-	1	3	-	2	-	-	1	-

Mapping of Course Outcomes with Program Specific Outcomes:

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

UNIT – I

Introduction to NoSQL: Definition and introduction, Traditional relational databases vs. NoSQL databases, Types of NoSQL databases, Storing and accessing data in NoSQL databases, Querying in NoSQL databases, Evolution and motivation behind NoSQL databases, Characteristics, advantages, and challenges of NoSQL databases, Performing CRUD operations, Creating records, Accessing data, Updating and deleting data.

Practice:

1. Software installation.
2. Write to connect to a MongoDB database and perform basic CRUD operations (Create, Read, Update, Delete) on a collection.

UNIT – II

Document Stores: Introduction to MongoDB, Importance of NoSQL databases, Data types, Documents, Nested documents, CRUD operations, Basic cursor methods: map, toArray, pretty, forEach, limit, count, sort, Columnar databases, Indexing and aggregation, MongoDB Node.js drivers and CAP theorem.

Practice:

1. Create a program to demonstrate indexing in MongoDB.
2. Develop a program to utilize MongoDB's Aggregation Framework.

UNIT – III

Key-Value Databases and Document Databases: What is a key-value store, Key-value store features, Suitable use cases, understanding key/value stores in Memcached and Redis, Eventually consistent non-relational databases, When not to use. What is a document database, Features, Suitable use cases, When not to use NoSQL databases.

Practice:

1. Write a program to create a simple key-value store that allows users to store and retrieve data using keys.
2. Implement functions to insert key-value pairs, retrieve values by key, update values, and delete key-value pairs.

UNIT – IV

NoSQL Stores and Data Evolution: Similarities between SQL and MongoDB query features, Accessing data from column-oriented databases like HBase, Querying Redis data stores, Changing document databases, Schema evolution in column-oriented databases, HBase data import and export, Data evolution in key/value stores.

Practice:

1. Design and create a column-oriented table schema in HBase.
2. Store and retrieve key/value pairs in Redis.
3. Perform basic data operations (SET, GET, DEL) and implement more complex operations.

UNIT – V

Indexing and Ordering Data Sets: Essential concepts behind a database index, Indexing and ordering in MongoDB, Creating and using indexes in MongoDB, Indexing and ordering in CouchDB, Indexing in Apache Cassandra.

Practice:

1. Create a collection and insert sample documents.
2. Create indexes on single fields, compound indexes (multiple fields), and multikey indexes (arrays).

Additional Practice:

1. Perform basic cursor Methods in MongoDB.
2. Study CAP theorem through real-world NoSQL examples.

Textbooks:

1. "NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence" – Pramod J. Sadalage, Martin Fowler, Addison-Wesley, ISBN: 978-0321826626.
2. "Making Sense of NoSQL" – Dan McCreary, Ann Kelly, Manning Publications, ISBN: 978-1617290651.

Reference Books:

1. "Professional NoSQL" – Shashank Tiwari, Wrox Press, Wiley, ISBN: 978-0-470-94224-0.
2. "Getting Started with NoSQL" – Gaurav Vaish, Packt Publishing, ISBN: 978-1782161315.

Web Links:

1. https://www.edx.org/learn/nosql
2. https://www.geeksforgeeks.org/dbms/introduction-to-nosql/
3. https://www.coursera.org/learn/nosql-databases?

INTERNET OF THINGS

Course Code: 2603MC34

L	T	P	C
2	0	2	4

Course Outcomes:

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

CO1: Able to understand the basic architecture of Internet of Things based Devices

CO2: Able to analyze light weight protocols implemented for connected devices

CO3: Able to develop firmware for controlling wireless devices

CO4: Able to design and develop Smart Devices using IoT

CO5: Able to evaluate and implement Smart Devices using IoT

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	PO9	PO10
CO1	2	-	3	-	3	-	-	3	-	-
CO2	2	-	3	-	3	-	-	3	-	-
CO3	2	-	3	1	3	-	-	3	-	-
CO4	2	-	3	1	3	-	-	3	-	-
CO5	-	-	3	1	3	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 IoT: What is the IoT and why is it important? Elements of an IoT ecosystem, Technology drivers, Business drivers, Trends and implications, Overview of Governance, Privacy and Security Issues.

Practice:

1. List 5 IoT devices you encounter daily (e.g., smart thermostat, fitness tracker).
2. Using one IoT example (e.g., smart home system), map the ecosystem:

Tools: Microcontroller (Arduino or ESP32), Required Sensors, Breadboard & jumper wires, Laptop with Arduino IDE, Free IoT cloud platform (ThingSpeak, Adafruit IO)

UNIT – II

IoT Protocols: Protocol Standardization for IoT – Efforts – M2M and WSN Protocols – SCADA and RFID Protocols – Issues with IoT Standardization – Unified Data Standards – Protocols – IEEE802.15.4–BACnet Protocol– Modbus – KNX – Zigbee– Network layer – APS layer – Security.

Practice:

1. Understand why protocol standardization is critical for IoT & Identify key characteristics of major IoT protocols
2. Distinguish between M2M, WSN, SCADA, and RFID protocols
3. Analyze IEEE 802.15.4, Zigbee layers (Network & APS), BACnet, Modbus, and KNX
4. Evaluate interoperability and standardization challenges

Tools: Microcontroller (Arduino or ESP32), Required Sensors, Breadboard & jumper wires, Laptop with Arduino IDE, Free IoT cloud platform (ThingSpeak, Adafruit IO), Cisco Packet Tracer (IoT mode), (Zigbee, IEEE 802.15.4 simulations), Tinkercad Circuits (optional for WSN concepts), Modbus Simulator (Modbus Poll / Modbus Slave), Wireshark (protocol inspection), Diagram Tool (draw.io / Lucidchart)

UNIT – III

IoT Architecture: IoT Open-source architecture (OIC)- OIC Architecture & Design principles- IoT Devices and deployment models - IoTivity: An Open source IoT stack - Overview- IoTivity stack architecture- Resource model and Abstraction.

Practice:

1. Explain the Open Interconnect Consortium (OIC) architecture
2. Understand OIC design principles (interoperability, scalability, security)
3. Describe the IoTivity open-source stack & Analyze the IoTivity stack architecture
4. Implement and interact with IoT resource models and abstraction

Tools: Microcontroller (Arduino or ESP32), Required Sensors, Breadboard & jumper wires, Laptop with Arduino IDE, Free IoT cloud platform (ThingSpeak, Adafruit IO), Cisco Packet Tracer (IoT mode), Tinkercad Circuits, Diagram Tool (draw.io / Lucidchart)

UNIT – IV

WEB OF THINGS: Web of Things versus Internet of Things – Two Pillars of the Web – Architecture Standardization for WoT– Platform Middleware for WoT – Unified Multitier WoT Architecture – WoT Portals and Business Intelligence.

Practice:

1. Understand architecture standardization for WoT
2. Identify the role of platform middleware in WoT
3. Design a unified multitier WoT architecture
4. Explore WoT portals and business intelligence dashboards

Tools: Web Browser, Node.js (WoT scripting), Postman / REST client, IoT/WoT simulator (Node-RED), Dashboard/BI Tool (Node-RED Dashboard / Grafana)

UNIT – V

IoT Applications: IoT applications for industry: Future Factory Concepts, Brownfield IoT, Smart Objects, Smart Applications. Study of existing IoT platforms /middleware, IoT- A, Hydra etc.

Practice:

1. Future Factory (Industry 4.0) concepts & UBrownfield IoT integration challenges
2. Identify smart objects and smart applications and compare IoT middleware platforms (IoT-A, HYDRA, etc.)
3. Design and simulate a simple industrial IoT solution
4. Analyze the role of middleware in industrial environments

Tools: Web Browser, Node-RED (industrial IoT simulation) and MQTT Broker (Mosquitto), Web Browser, Node-RED (industrial IoT simulation), MQTT Broker (Mosquitto), Postman / REST client, Dashboard tool (Node-RED Dashboard / Grafana), Diagram tool

Additional Practice:

1. Core (Must-Have)
2. Node-RED
3. ESP32 / Arduino
4. MQTT Broker (Mosquitto)

Textbooks:

1. David Hanes, Gonzalo Salgueiro, Patrick Grossetete, Robert Barton, Jerome Henry,"IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things", First Edition, Cisco Press, 2017. ISBN: 978-9386873743 / 978-1587144561.
2. Adrian Mcewen, HakinCassimally, "Designing The Internet of Things", First Edition, Wiley, 2014. ISBN-10: 111843062X / ISBN-13: 978-1-118-43062-0
3. Honbo Zhou, "The Internet of Things in the Cloud: A Middleware Perspective", CRC Press, 2012. ISBN-13 (print): 978-1-4398-9299-2 / ISBN-10 (print): 1439892997
4. Olivier Hersent, David Boswarthick, Omar Elloumi , "The Internet of Things – Key Applications and Protocols", Wiley, 2012, ISBN-13: 978-1-119-99435-0 / ISBN-10: 1119994357.

Reference Books:

1. Javed, A. (2016). Building Arduino projects for the Internet of Things: Experiments with real-world applications. Apress. https://doi.org/10.1007/978-1-4842-1940-9, ISBN-13: 9781484219393.
2. Raj Kamal, "Internet of Things: Architecture and Design Principles",First Edition, McGraw Hill Education, 2017. ISBN-13: 978-93-5260-522-4 / ISBN-10: 93-5260-522-5.
3. Peter Waher, "Mastering Internet of Things: Design and create your own IoT applications using Raspberry Pi 3", First Edition, Packt Publishing, 2018. ISBN-13: 978-1-78839-748-3 / ISBN-10: 1788397487

Web Links:

1. https://arduinogetstarted.com/tutorials/arduino-iot?utm_source=chatgpt.com
2. https://iot4beginners.com/?utm_source=chatgpt.com
3. https://hackr.io/tutorials/learn-internet-of-things-iot?utm_source=chatgpt.com
4. https://www.jitimu.com/arduino-stem-projects/iot-tutorials/?utm_source=chatgpt.com

DATA VISUALIZATION

Course Code: 2603MC35

L	T	P	C
3	0	1	4

Course Outcomes:

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

CO1: Understand fundamental concepts of data visualization and visual analytics using Tableau.

CO2: Create effective visualizations using appropriate chart types and interactive features.

CO3: Apply calculations, aggregations, and table calculations to enhance data analysis.

CO4: Use geographic visualization techniques to derive spatial insights from data.

CO5: Develop interactive, ad-hoc analytical dashboards and forecasting solutions using Tableau Server/Desktop.

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	PO9	PO10
CO1	3	2	1	3	1	2	-	-	1	2
CO2	2	3	2	2	2	2	-	-	2	2
CO3	3	3	2	3	1	2	-	-	2	2
CO4	2	2	2	3	1	2	-	2	-	2
CO5	2	3	3	3	2	3	-	2	-	3

Mapping of Course Outcomes with Program Specific Outcomes:

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

UNIT – I

Introduction to Data Visualization & Tableau: Data Visualization concepts, Visual Analytics, Tableau Desktop, connecting to data, generated fields, direct vs extract connections, joining tables, blending data sources.

Practice:

1. Connect Tableau to CSV and Excel data sources and compare live vs extract connections.
2. Perform inner, left, and full joins on sample datasets.
3. Blend two heterogeneous data sources and create a simple worksheet.

UNIT – II

Building Visualization: Chart types, text tables, maps, bar, line, area, pie charts, scatter plots, bullet graphs, Gantt charts, sorting, filters, sets, groups, hierarchies.

Practice:

1. Create at least five different chart types for a given business dataset.
2. Apply filters, groups, and hierarchies to analyze sales by region and category.
3. Design a simple dashboard combining multiple views.

UNIT – III

Calculations and Analytics: Aggregation concepts, calculated fields, table calculations, calculation dialog box, table calculation functions.

Practice:

1. Create calculated fields for profit margin and growth rate.
2. Implement running totals, moving averages, and percent of total using table calculations.
3. Compare aggregated vs disaggregated results.

UNIT – IV

Geographic Visualization: Standard maps, plotting custom locations, replacing default maps, shaping data for point-to-point mapping.

Practice:

1. Build a geographic map using latitude and longitude.
2. Customize map layers and replace base maps.
3. Create a point-to-point flow map for logistics or migration data.

UNIT – V

Ad-hoc Analysis & Advanced Features: Forecasting, parameters for ad-hoc analysis, editing views on Tableau Server.

Practice:

1. Generate forecasts for time-series sales data.
2. Create interactive dashboards using parameters.
3. Publish and edit views on Tableau Server (or simulated environment).

Textbooks:

1. "Tableau your data" - Daniel G. Murray and the Inter works BI team, 2nd Edition (2016), Wiley Publications, ISBN-13: 978-1119001195.
2. "Storytelling with Data: A Data Visualization Guide for Business Professionals" - Cole Nussbaumer Knaflic, 1st Edition (2015), ISBN-13: 978-1119002253.

Reference Books:

1. "Visualization Analysis & Design" - Tamara Munzner, 1st Edition, A K Peters Visualization Series, ISBN-10: 1466508190.
2. "Interactive Data Visualization for the Web" - Scott Murray, 2nd Edition, MGH, ISBN-10: 1491921288.

Web Links:

1. https://elearn.nptel.ac.in/shop/iitworkshops/completed/datavisualizationwithr/?v=c86ee0d9d7ed
2. https://www.geeksforgeeks.org/datavisualizationanditsimportance/

BUSINESS INTELLIGENCE & ANALYTICS

Course Code: 2603MC36

L	T	P	C
3	0	1	4

Course Outcomes:

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

CO1: Explain the concepts, architecture, and importance of Business Intelligence and Business Analytics in organizational decision making.

CO2: Understand and analyze decision-making processes and various types of Decision Support Systems.

CO3: Interpret business data using basic analytical and predictive techniques.

CO4: Apply model-based decision-making approaches under certainty, uncertainty, and risk.

CO5: Describe automated decision systems and expert systems and assess their applications in business environments.

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	PO9	PO10
CO1	3	2	2	3	-	3	2	2	-	2
CO2	2	3	3	3	2	2	-	2	-	2
CO3	3	3	2	3	-	2	-	-	2	3
CO4	2	3	3	3	2	2	-	2	-	3
CO5	2	-	2	3	-	2	-	-	-	3

Mapping of Course Outcomes with Program Specific Outcomes:

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

UNIT – I

Introduction to Business Intelligence and Analytics: Overview of Business Intelligence (BI) and Business Analytics (BA); information systems for decision making; evolution of decision support systems; BI framework and architecture; role of BI in organizations; introduction to big data analytics.

Practice:

1. Power BI/Tableau dashboard creation
2. Pivot tables in Excel

3. BI vs. BA dataset comparison

UNIT – II

Decision Making and Decision Support Systems: Decision making concepts and types; phases of decision-making process – intelligence, design, choice, and implementation; decision support systems capabilities; DSS classification; components and architecture of DSS.

Practice:

1. Loan approval DSS in Excel
2. Supplier selection matrix
3. DSS simulation in Google Sheets

UNIT – III

Analytics Techniques for Business: Introduction to descriptive, predictive, and prescriptive analytics; basics of neural networks; support vector machines; nearest neighbor methods; overview of sentiment analysis and its business applications.

Practice:

1. Descriptive analytics in Python
2. Regression model for prediction
3. Nearest neighbor classifier
4. SVM classification
5. Sentiment analysis on Twitter data

UNIT – IV

Model-Based Decision Making: Decision models and DSS modeling; certainty, uncertainty, and risk; spreadsheet-based decision modeling; optimization concepts; decision tables; decision trees.

Practice:

1. Decision tree in Excel
2. Optimization with Excel Solver
3. Decision table in Sheets

UNIT – V

Automated Decision Systems and Expert Systems: Automated decision systems; overview of artificial intelligence; expert systems concepts; structure and components of expert systems.

Practice:

1. Rule-based expert system in Python
2. Chatbot with ChatterBot
3. AI-powered recommendation system
4. Expert system flowchart in Excel

Textbooks:

1. "Business Intelligence and Analytics: Systems for Decision Support" - Ramesh Sharda, Dursun Delen, Efraim Turban, J. E. Aronson,Ting Peng Liang, David King, 10th Edition, Pearson, ISBN-13: 978-0133050905.
2. "Business Intelligence, Analytics, and Data Science" - Ramesh Sharda, 4th Edition, Pearson, ISBN-13: 978-0134633282.

Reference Books:

1. "Analytics at Work: Smarter Decisions, Better Results" - Thomas H. Davenport, Jeanne G. Harris, 1st Edition (2010), Harvard Business Review Press, ISBN-13: 978-1422177693.
2. "Practical Business Intelligence with Power BI" - Devin Knight, Brian Knight, 1st Edition (2018), Packt Publishing, ISBN-13: 978-1788290147.
3. "Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython" - Wes McKinney, 2nd Edition (2017), O'Reilly Media, ISBN-13: 978-1491957660.

Web Links:

1. NPTEL – Business Intelligence and Analytics (https://www.youtube.com/watch?v=f1if5sqCS4U)
2. Power Bi tutorial for Beginners | Power BI Dashboard Project in Hindi | Power Bi Course 2023
3. Data Analysis with Python - Full Course for Beginners (Numpy, Pandas, Matplotlib, Seaborn)

UNIVERSAL HUMAN VALUES

Course Code: 2603MC41

L	T	P	C
2	0	0	2

Course Outcomes:

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

CO1: Make use of self-exploration, continuous happiness and prosperity, right understanding, and relationship to live in harmony principles at all levels.

CO2: Explore the harmony in the human being and harmony in self.

CO3: Interpret the harmony in the family and society and harmony in human–human relationship using justice, truth, respect, prosperity, and co-existence.

CO4: Appraise the harmony in nature and existence using interconnectedness and mutual fulfillment, co-existence, and holistic perception of harmony at all levels of existence.

CO5: Apply the principles of holistic understanding of harmony in professional ethics to improve competence.

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	PO9	PO10
CO1	-	-	-	-	1	1	2	-	-	2
CO2	-	-	-	-	2	-	1	-	-	2
CO3	-	-	-	-	2	2	2	3	1	1
CO4	-	-	-	-	-	1	1	3	-	1
CO5	-	-	-	-	2	2	3	2	2	2

Mapping of Course Outcomes with Program Specific Outcomes:

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

UNIT – I

Course Introduction – Need, Basic Guidelines, Content and Process for Value Education

1. Purpose and motivation for the course, recapitulation from Universal Human Values.
2. Self-Exploration – what is it? Its content and process; Personality Traits – Self Excellence, Natural Acceptance and Experiential Validation as the process for self-exploration, adaptability, belief and understanding; Self-discipline.

3. Continuous Happiness and Prosperity – A look at basic human aspirations.
4. Right understanding, Relationship and Physical Facility – the basic requirements for fulfillment of aspirations of every human being with their correct priority.
5. Understanding Happiness and Prosperity correctly – A critical appraisal of the current scenario.
6. Method to fulfill the above human aspirations – understanding and living in harmony at various levels.
7. Myers–Briggs Type Indicator (MBTI) Personality test.

UNIT – II

Understanding Harmony in the Human Being – Harmony in Myself

1. Understanding human being as a co-existence of the sentient 'I' and the material 'Body'.
2. Understanding the needs of Self ('I') and 'Body' – happiness and physical facility.
3. Understanding the Body as an instrument of 'I' (I being the doer, seer and enjoyer); Habits and Hobbies; SWOT Analysis (Activity).
4. Understanding the characteristics and activities of 'I' and harmony in 'I' – Dalai Lama's Tibetan Personality Test – Dr. Menninger's Psychometric Test.
5. Understanding the harmony of 'I' with the Body – Sanyam and Health; correct appraisal of physical needs; meaning of Prosperity in detail.
6. Programs to ensure Sanyam and Health.
7. Epidemiology – Definition of health; Social and Preventive Medicine; Personal hygiene and handling stress; WHO Guidelines.

UNIT – III

Understanding Harmony in the Family and Society – Harmony in Human–Human Relationship

1. Understanding values in human–human relationship; meaning of Justice (nine universal values in relationships) and program for its fulfillment to ensure mutual happiness; Trust and Respect as the foundational values of relationship.
2. Understanding the meaning of Trust; difference between intention and competence.
3. Understanding the meaning of Respect; difference between respect and differentiation; other salient values in relationship – Friends and Foes, Empathy, False Prestige.
4. Concept of an Ideal family – Marriage as an Institution.
5. Understanding the harmony in the society (society being an extension of family): Resolution, Prosperity, Fearlessness (Trust), and Co-existence as comprehensive human goals.
6. Visualizing a universal harmonious order in society – Undivided Society, Universal Human Order – from family to world family.

UNIT – IV

Understanding Harmony in the Nature and Existence – Whole Existence as Coexistence

1. Understanding the harmony in nature and its equanimity; respect for all; nature as a teacher.
2. Interconnectedness and mutual fulfillment among the four orders of nature; recyclability and self-regulation in nature.
3. Understanding existence as co-existence of mutually interacting units in all-pervasive space.
4. A holistic perception of harmony at all levels of existence.

UNIT – V

Implications of the above Holistic Understanding of Harmony on Professional Ethics

1. Natural acceptance of human values.
2. Definitiveness of Ethical Human Conduct.
3. Basis for Humanistic Education, Humanistic Constitution, and Humanistic Universal Order.
4. Competence in professional ethics:
   a. Ability to utilize professional competence for augmenting universal human order.
   b. Ability to identify the scope and characteristics of people-friendly and eco-friendly production systems.
   c. Ability to identify and develop appropriate technologies and management patterns for the above production systems.
5. Case studies of typical holistic technologies, management models, and production systems.
6. Vision for the holistic alternatives; UHV's for entrepreneurship.
7. Strategy for transition from the present state to Universal Human Order:
   a. At the level of individual as socially and ecologically responsible engineers, technologists and managers.
   b. At the level of society as mutually enriching institutions and organizations; right understanding and dilemmas of professional ethics in today's world.

Textbooks:

1. Human Values, A. N. Tripathi, New Age International Publishers, New Delhi, ISBN-10: 8122425895.
2. Human Values and Professional Ethics, R. R. Gaur, R. Sangal, G. P. Bagaria, Excel Books, New Delhi, ISBN-10: 9788174467812.

Reference Books:

1. Jeevan Vidya: Ek Parichaya, A. Nagaraj, Jeevan Vidya Prakashan, Amarkantak, ISBN: 978-93-87034-53-2.
2. Hind Swaraj or Indian Home Rule, Mohandas K. Gandhi, ISBN: 978-1449922214.
3. India Wins Freedom, Maulana Abul Kalam Azad, ISBN: 9789355496218.

Web Links:

1. https://www.youtube.com/watch?v=OgdNx0X923I&list=PLYwzG2fd7hzc4HerTNkc3pS_IvcCfKznV

ENVIRONMENTAL SCIENCE & SUSTAINABILITY

Course Code: 2603MC42

L	T	P	C
2	0	0	2

Course Outcomes:

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

CO1: Understand the scope of Environmental Studies, sustainability concepts, and the structure and functioning of ecosystems.

CO2: Analyze natural resources, land-use changes, water and energy issues, and their environmental impacts.

CO3: Identify types of pollution, their causes, effects, control measures, and waste management practices.

CO4: Explain environmental problems, environmental laws, and international environmental agreements.

CO5: Assess the effects of population growth, disaster management practices, environmental movements, and environmental 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	PO9	PO10
CO1	1	1	-	-	-	-	-	-	-	1
CO2	1	2	-	-	-	-	-	3	-	-
CO3	-	-	-	-	-	-	-	-	-	-
CO4	-	1	-	-	-	-	2	-	-	-
CO5	-	-	-	-	1	1	2	-	-	2

Mapping of Course Outcomes with Program Specific Outcomes:

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

UNIT – I

Nature of Environmental Studies: Scope and Importance - Concept of sustainability and Sustainable Development. Eco systems- Meaning - Structure and function of ecosystem - Energy flow in an ecosystem - food chains, food webs and ecological succession - Forest ecosystem - Grassland ecosystem - Desert ecosystem - Aquatic ecosystems (ponds, streams, lakes, rivers, oceans, estuaries).

UNIT – II

Renewable and Non-renewable Resources: Land resources and land use change; Land degradation, soil erosion and desertification - Deforestation: Causes and impacts due to mining, dam building on environment, forests - biodiversity and tribal populations - Water: Use and over-exploitation of surface and ground water, floods, droughts, conflicts over Water (inter-state).-Energy resources: Renewable and non-renewable energy sources, use of alternate energy sources, growing energy needs.

UNIT – III

Environmental pollution: types, causes, effects and controls; Air, water, soil and noise pollution.- Nuclear hazards and human health risks. - Solid waste management: Control measures of urban and industrial waste. -Pollution case studies.

UNIT – IV

Environmental Policies & Practices: Climate change, global warming, ozone layer depletion, acid rain and impacts on human communities and agriculture 2/2 - Environment Laws: Environment Protection Act; Air (Prevention & Control of Pollution) Act; Water (Prevention and control of Pollution) Act; Wildlife Protection Act; Forest Conservation Act. International agreements: Montreal and Kyoto protocols and Convention on Biological Diversity (CBD).

UNIT – V

Human population growth: Impacts on environment, human health and welfare. - Resettlement and rehabilitation of project affected persons; case studies. - Disaster management: floods, earthquake, cyclones and landslides. - Environmental movements: Chipko, Silent valley, Bishnois of Rajasthan-Environmental ethics: Role of Indian and other religions and cultures in environmental conservation.

Textbooks:

1. Cunningham, W.P. and Saigo, B.W., 1995. Environmental Science, 3rd Edition. W.M.C. Brown Publishers, New York, USA, ISBN-10: 0697158942.
2. Enger, D.E. and Smith B.F., 1995. Environment Science–A Study of Interrelationships, 5th Edition, W.M.C. Brown Publishers, New York, USA, ISBN-10: 0697295869

Reference Books:

1. Negi, B.S., 1991, Geography of Resources, 2nd edition Kedar Nath Ram Nath Meerut. ISBN-13: 978-9383684335
2. Trivedi, P.R., 1999, Encyclopedia of Ecology and Environment, 1-10, Indian Institute of Ecology and Environment, 1st edition, New Delhi ISBN: 8174560548

Web Links:

1. https://syllabiref.sukantamahavidyalaya.ac.in/home/envs?utm_source=chatgpt.com
2. https://www.haldiagovtcollege.ac.in/environmental-studies-aecc/?utm_source=chatgpt.com

INDIAN KNOWLEDGE SYSTEMS

Course Code: 2603MC43

L	T	P	C
2	0	0	2

Course Outcomes:

At the end of the Course, Student 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.

CO5: Evaluate the relevance and protection of Traditional Knowledge in modern 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	PO9	PO10
CO1	1	1	-	-	-	-	-	-	-	2
CO2	1	2	-	-	-	-	1	1	-	-
CO3	-	-	-	-	-	-	2	2	-	-
CO4	-	1	1	-	-	-	2	3	-	-
CO5	-	2	-	-	-	-	3	2	-	1

Mapping of Course Outcomes with Program Specific Outcomes:

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

Textbooks:

1. Kapil Kapoor & Avadhesh Kumar Singh, Indian Knowledge Systems, D.K. Printworld (India) Pvt. Ltd., New Delhi. ISBN: 9788124608177.
2. S. Radhakrishnan, Indian Philosophy – Volume I & II, Oxford University Press, New Delhi. ISBN: 9780195698418.
3. O.P. Jaggi, Science and Technology in Ancient India, Atma Ram & Sons, Delhi. ISBN: 9788185694085.
4. Madhukar K. Bhagat, Indian Heritage and Culture, McGraw-Hill Education (India). ISBN: 9789332901476.
5. Yogendra K. Malik, Traditional Knowledge System in India, Pearson Education India. ISBN: 9788131762219.

Reference Books:

1. Debiprasad Chattopadhyaya, Ancient Indian Science and Technology, History of Science, Philosophy and Culture in Indian Civilization Series. ISBN: 9788187586098.
2. Nitin Singhania, Indian Culture and Heritage, McGraw-Hill Education. ISBN: 9789353161132.
3. Bhu Dev Sharma, Knowledge Traditions of India, Indian Council of Philosophical Research (ICPR). ISBN: 9788185636306.

Web Links:

1. https://iks.aicte-india.org/
2. https://www.education.gov.in/iks

INDIAN CONSTITUTION

Course Code: 2603MC44

L	T	P	C
2	0	0	2

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	PO9	PO10
CO1	1	1	-	-	-	-	-	-	-	2
CO2	1	2	-	-	-	-	1	1	-	-
CO3	-	-	-	-	-	-	2	2	-	-
CO4	-	1	1	-	-	-	2	3	-	-
CO5	-	2	-	-	-	-	3	2	-	1

Mapping of Course Outcomes with Program Specific Outcomes:

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

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 PanchayatiRaj: 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.

Textbooks:

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/

FUNDAMENTAL ENGLISH PROFICIENCY

Course Code: 2603MC37

L	T	P	C
0	0	1	1

Course Outcomes:

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

CO1: Understand 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, rhythm, 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	PO9	PO10
CO1	-	-	-	-	-	2	-	-	-	-
CO2	-	-	-	-	-	3	-	-	-	-
CO3	-	-	-	-	-	2	-	-	-	-
CO4	-	-	-	-	-	3	-	-	-	-
CO5	-	-	-	-	-	1	-	-	-	-

Practice:

UNIT – 1 **Outstanding people A/B/C/D**

Listening	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 – 2	**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 – 3	**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 – 4	**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 – 5	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

Suggested Software:

- Cambridge Empower
- Soft X (K-Van Solutions)

Textbooks:

1. Cambridge Empower – Second Edition B2 Level - Adrian Doff, Craig Thaine, HerbertPuchta, Jeff Stranks, Peter Lewis – Jones.

Reference Books:

1. M Ashraf Rizvi: Effective Technical Communication
2. Raymond Murphy: English Grammar in Use, Cambridge University Press. Fifth Edition
3. J. Sethi & P.V. Dhamija. A Course in Phonetics and Spoken English, (2nd Ed), Kindle

Web links:

1. www.cambridgeone.org
2. https://www.britishcouncil.in/english/online
3. www.englishmedialab.com

ADVANCED ENGLISH PROFICIENCY

Course Code: 2603MC38	**L**	**T**	**P**	**C**
	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	PO9	PO10
CO1	-	-	-	-	-	2	-	-	-	-
CO2	-	-	-	-	-	3	-	-	-	-
CO3	-	-	-	-	-	2	-	-	-	-
CO4	-	-	-	-	-	3	-	-	-	-
CO5	-	-	-	-	-	1	-	-	-	-

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

Suggested Software:

- Cambridge Empower
- Soft X (K-Van Solutions)

Textbooks:

1. Cambridge Empower – Second Edition B2 Level - Adrian Doff, Craig Thaine, Herbert Puchta, Jeff Stranks, Peter Lewis – Jones.

Reference Books:

1. Raman Meenakshi, Sangeeta-Sharma. Technical Communication. Oxford University Press.
2. Michael Swan- Practical English Usage
3. Taylor Grant: English Conversation Practice, Tata McGraw-Hill Education India.

Web Links:

1. https://www.coursera.org/
2. https://www.skillshare.com/
3. https://www.mindtools.com/cawh8bu/communication-tools

EMPLOYABILITY SKILLS - I

Course Code: 2603MC45

L	T	P	C
0	0	1	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	PO9	PO10
CO1	3	2	2	1	-	-	-	-	-	2
CO2	2	3	1	1	-	-	-	-	-	2
CO3	2	1	2	1	-	3	-	-	-	2
CO4	-	1	-	-	1	-	2	1	-	2

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

Textbooks:

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

Course Code: 2603MC46	**L**	**T**	**P**	**C**
	0	**0**	**1**	**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	PO9	PO10
CO1	3	3	2	1	-	-	-	-	-	2
CO2	2	3	1	1	-	-	-	-	-	2
CO3	2	1	2	1	-	-	-	-	-	2
CO4	1	2	2	1	-	-	-	-	-	2
CO5	-	1	-	-	2	3	2	1	-	2

Aptitude: 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

Textbooks:

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

Course Code: 2603MC47

L	T	P	C
0	0	1	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	PO9	PO10
CO1	3	3	2	1	-	-	-	-	-	2
CO2	2	3	1	1	-	-	-	-	-	2
CO3	2	1	2	1	-	-	-	-	-	2
CO4	1	2	2	1	-	-	-	-	-	2
CO5	-	1	-	-	2	3	2	1	-	2

Aptitude: 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

Textbooks:

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

Course Code: 2603MC48

L	T	P	C
0	0	1	1

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	PO9	PO10
CO1	3	3	2	1	-	-	-	-	-	2
CO2	2	3	1	1	-	-	-	-	-	2
CO3	2	1	2	1	-	-	-	-	-	2
CO4	1	2	2	1	-	-	-	-	-	2
CO5	-	1	-	-	2	3	2	1	-	2

Aptitude: 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

Textbooks:

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

DESIGN THINKING & INNOVATION

Course Code: 2603MC40

L	T	P	C
0	0	2	2

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:

CO/PO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO9	PO10
CO1	-	3	2	-	-	-	-	-	-	-
CO2	-	1	3	2	-	-	-	-	-	-
CO3	-	2	-	1	-	-	-	-	-	-
CO4	-	-	1	3	-	2	-	-	-	-
CO5	-	-	2	-	2	3	-	-	-	-

Mapping of Course Outcomes with Program Specific Outcomes:

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

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

UNIT – III

Design Thinking in IT: Design Thinking to Business Process modelling – Agile in Virtual collaboration environment – Scenario based Prototyping

UNIT – IV

DT For strategic innovations: Growth – Story telling representation – Strategic Foresight - Change – Sense Making - Maintenance Relevance – Value redefinition - Extreme Competition – experience design.

UNIT – V

Design thinking workshop: Design Thinking Workshop Empathize, Design, Ideate, Prototype and Test.

Textbooks:

1. “Engineering Design”- John.R.Karsnitz, Stephen O’Brien and John P. Hutchinson Second Edition, Cengage learning (International edition), ISBN Number: 978-1-305- 49650-1
2. "The Design of Business: Why Design Thinking is the Next Competitive Advantage" - Roger Martin, First Edition, Harvard Business Press, ISBN Number: 978-1422138565

Reference Books:

1. “Engineering Design Process” - Yousef Haik and Tamer M.Shahin, Second Edition, Cengage Learning, ISBN Number: 978-1-305-20680-4
2. Solving Problems with Design Thinking - Ten Stories of What Works - Hardcover by Jeanne Liedtka, Andrew King, Kevin Bennett, Columbia Business School Publishing, ISBN Number: 978-0231156366.

Web Links:

1. www.tutor2u.net/business/presentations/
2. www./Product lifecycle/default.html
3. https://docs.oracle.com/cd/E11108_02/otn/pdf /