DR. BABASAHEB AMBEDKAR MARATHWADA UNIVERSITY, CHHATRPATI SAMBHAJINAGR.

CIRCULAR NO.SU/NEP/B.Sc. Honor's/Model College/38/2024

It is hereby inform to all concerned that, the syllabi prepared by the Ad-hoc Boards and recommended by the Dean, Faculty of Science & Technology, the Hon'ble Vice-Chancellor has accepted the **following syllabi under National Education Policy-2020 as per Guidelines of UGC** run at Model College, Ghansawangi Dist.-Jalna in his emergency powers under section 12(7) of the Maharashtra Public Universities Act, 2016 on behalf of the Academic Council as appended herewith.

Sr.No.	Courses	Semester
1.	Honours Degree of B.Sc. Biotechnology	IIIrd & IVth
2.	Honour's Degree of B.Sc. Biochemistry	IIIrd & IVth
3.	Honour's Degree of B.Sc. Computer Science	IIIrd & IVth

This is effective from the Academic Year 2024-25 and onwards.

All concerned are requested to note the contents of this circular and bring the notice to the students, teachers and staff for their information and necessary action.

University Campus,
Chhatrapati Sambhajinagar
431 004.
REF.NO.SU/2024/ 7081-89
Date:- 10.09.2024

★★★★★

Deputy Registrar,
Academic Section

Copy forwarded with compliments to :-

1] **The Principal, Model College, Ghansawangi Dist. Jalna**
Dr. Babasaheb Ambedkar Marathwada University,

2] **The Director, University Network & Information Centre, UNIC, with a request to upload this Circular on University Website.**

Copy to :-

1] **The Director, Board of Examinations & Evaluation,** Dr.Babasaheb Ambedkar Marathwada University, Chhatrapati Sambhajinagar.

2] The Section Officer,[B.Sc.Unit] Examination Branch, Dr.Babasaheb Ambedkar Marathwada University, Chhatrapati Sambhajinagar.

3] The Programmer [Computer Unit-1] Examinations, Dr.Babasaheb Ambedkar Marathwada University, Chhatrapati Sambhajinagar.

4] The Programmer [Computer Unit-2] Examinations, Dr.Babasaheb Ambedkar Marathwada University, Chhatrapati Sambhajinagar.

5] The In-charge,[E Suvidha Kendra], Rajarshi Shahu Maharaj Pariksha Bhavan, Dr.Babasaheb Ambedkar Marathwada University, Chhatrapati Sambhajinagar.

6] The Public Relation Officer, Dr.Babasaheb Ambedkar Marathwada University, Chhatrapati Sambhajinagar.

7] The Record Keeper, Dr.Babasaheb Ambedkar Marathwada University, Chhatrapati Sambhajinagar.

-=**=-

Dr. Babasaheb Ambedkar Marathwada University,
Chhatrapati Sambhajinagar.

Structure and Syllabus for B.Sc. (Computer Science)

(Four Years Multidisciplinary Degree Program with Multiple Entry and Exit Option)

FOUR YEAR BACHELOR OF SCIENCE (B.SC)
COMPUTER SCIENCE

(For Model College Ghansawangi Dist. Jalna)

Under the Faculty of
Science and Technology

Effective from Academic year 2024 – 2025

(As per NEP-2020)

Dean
Faculty of Science & Technology
Dr. Babasaheb Ambedkar Marathwada
University,

1. Preamble

Education is the key to development of any society. Role of higher education is crucial for securing right kind of employment and also to pursue further studies in best institutes elsewhere within and outside India.

Quality of higher education in particular deserves high priority to enable the young and future generation of students. This curriculum is to acquire skill, training , knowledge in order to enhance their thinking, creativity, comprehension and application abilities for prepare them to compete, succeed and excel globally.

Sustained initiatives are required to reform the present higher education system for improving and upgrading the academic resources and learning environments.

The Undergraduate Curriculum of National Education Policy 2020 underlines the historical perspective, philosophical basis, and contemporary realities of higher education.

The resultant outcome of this comprehensive exercise undertaken by this curriculum is not only underlines the heart and soul of the NEP 2020 and spirit but also goes on to create a teaching-learning framework at the undergraduate level to attract the young minds towards research, innovation, apprenticeship, social outreach, entrepreneurship and similar such areas of human knowledge and endeavor while imbibing the truly charged academic environ of college life.

The following objectives of NEP are kept in perspective while framing this curriculum.

- To promote each student's holistic development in both academic and non-academic spheres
- To provide flexibility to students so that learners have the ability to choose their learning trajectories and programs, and thereby choose their paths in life according to their talents and interests.
- Multidisciplinary and holistic education to ensure the unity and integrity of all knowledge thereby eliminating harmful silos between different areas of learning
- To promote creativity and critical thinking and to encourage logical decision-making and innovation;
- To promote ethics and human & Constitutional values;
- To promote multilingualism and the power of language in learning and teaching;
- To impart life skills such as communication, cooperation, teamwork, and resilience;

- To promote outstanding research as a corequisite for outstanding education and development.

The below mentioned objectives have been reflected in various features in this curriculum.

1. **Holistic development of the students**
2. **Flexibility**
3. **Multiple entry and exit facility**
4. **Academic bank of credits**
5. **Multilingualism**
6. **Research component**

2. Abbreviations

1. **GE/OE:** Generic / Open Elective
2. **VSEC:** Vocational Skill and Skill Enhancement Course
3. **VSC:** Vocational Skill Courses
4. **SEC :** Skill Enhancement Courses
5. **AEC :**Ability Enhancement Course
6. **IKS:** Indian Knowledge System
7. **VEC:** Values Education Courses
8. **OJT:** On job Training: Internship/ Apprenticeship
9. **FP:** Filed Project
10. **CEP:** Community Engagement and Service
11. **CC:** Co-Curricular Courses
12. **RM:** Research Methodology
13. **RP:** Research Project

Definitions of Key Words:

a. Academic Year: Two consecutive (one odd + one even) semesters constitute one academic year

b. Course: Usually referred to as 'paper', which is a component of a program. All courses need not carry the same weight. The courses should define learning objectives and learning outcomes. A course may be designed to comprise lectures/ tutorials/laboratory work/ field work/ project work/ vocational training/viva/ seminars/term papers / assignments / presentations/ self-study etc. or a combination of some of these.

c. Credit: A unit by which the course work is measured. It determines the number of hours of instructions required per week in a semester. One credit is equivalent to one hour of lecture or tutorial or two hours of practical work/field work per week in a semester.

d. Grade Point: It is a numerical weight allotted to each letter grade on a 10-point scale.

e. Credit Point: It is the product of grade point and number of credits for a course.

f. Letter Grade: It is an index of the performance of students in a said course. Grades are denoted by letters O, A+, A, B+, B, C, P and F.

g. Program: It is a study in a discipline leading to award of a Degree, diploma or certificate.

h. Semester: Each semester will consist of over 16 weeks of academic work equivalent to 90 actual teaching days. The odd semester may be generally scheduled from June to November and even semester from January to May.

i. Semester Grade Point Average (SGPA): It is a measure of performance of work done in a semester. It is the ratio of total credit points secured by a student in various courses registered in a semester and the total course credits taken during that semester. It shall be expressed up to two decimal places

j. Cumulative Grade Point Average (CGPA): It is a measure of overall cumulative performance of a student over all the semesters of a program. The CGPA is the ratio of total credit points secured by a student in various courses in all the semesters and sum of the total credits of all courses in all the semesters. It is expressed up to two decimal places.

3. Concepts:

1. Major Subject: A major subject/core subject is the subject that represents the main focus of a degree. A degree in major is permitted if the student opts and accrues courses with minimum of 50% credits corresponding to the major.

2. Minor Subject: A minor is secondary subject that may complement the major or can have interdisciplinary bandwidth. Minors are a bunch of courses and can be related or unrelated to the major. A university must declares the minors and make it available to the students to choose from.

3. Internship: Students at all HEIs will provided with opportunities for internship in local industries, businesses, banks, artists, crafts persons, and so on. As well as research internships with faculty and researchers at their own or other HEIs/research institution.

4. Generic/Open Elective (OE): Generic or Open Elective shall be a pool/ basket of courses which is meant to provide multidisciplinary or interdisciplinary education to students. GEs or Ses shall consist of pool/ basket of courses offered by the various departments under different discipline/faculty of study.

5. Ability Enhancement Courses (AEC) - Language, Literature & Environment Studies. The courses based upon the content that leads to knowledge enhancement through various areas of study. They are language and literature and environment science and sustainable development which will be mandatory for all disciplines.

6. Indian Knowledge System (IKS): IKS is a generic phrase that covers practically everything about India. While designing the curriculum, it need to be understood that IKS is not about merely knowing some ancestral knowledge. IKS is also about protecting received wisdom, economic security, and national pride.

7. Vocational and Skill Enhancement Courses (VSEC): In all disciplines/faculties are aimed at providing hands-on-training, competencies, proficiencies and skills to students. The basic purpose of vocational courses is to enhance skill and employability.

8. Internship and apprenticeship: It provides for an industry led, practice oriented and outcome based learning. Forming robust industry-academia linkage. Every HEI will undertake MOUs with industry for industry-institute linkage for promotion of apprenticeship/ Internship/ research/ entrepreneurship/ employment opportunities. There will be 4-6 weeks of structured summer internship in industry or research institute.

9. Field Projects: The field based learning/ projects should attempt to provide opportunities for students to understand the different socio-economic contexts. It should aim at giving student exposure to development-related issues in rural and urban settings. It will provide opportunities for the student to observe situation in rural and urban contexts, and to observe and study actual field situation regarding issues related to socio-economic development.

10. Community engagement and service: Community engagement and service seeks to expose students to the socio-economic issues in society so that the theoretical learning can be supplemented by actual life experiences to generate solutions to real life problems. This component will include participation in activities related to NSS, NCC, adult education/ literacy initiatives and mentoring school students.

(Note: Change in Structure will be applicable as per guidelines of University Grant Commission and State Government of Maharashtra released from time to time)

B. Sc. Computer Science First Year (Semester III) (Level 5.0)
Teaching Scheme

Year/ Semester and Level		Section	Course Code	Course Name	Credits Assigned			Teaching Scheme (Hrs/ week)	
					Theory	Practical	Total	Theory	Practical
Second Year Semester III Level 4.5	Major	DSE-5	NCOM301T	Database Management System	03	--	03	03	--
			NCOM301P	Lab based on Database Management System	...	02	02	--	04
		DSE-6	NCOM302T	Problem Solving using Java Language	03	--	03	03	--
			NCOM302P	Java programming Lab	---	02	02	---	04
	Supportive	**Minor-3** Select any one course from Bucket 1	NCOM303T1	Computational Ethics	02	--	02	02	--
			NCOM303T2	Advance Excel	02	--	02	02	--
			NCOM303T3	Internet Technology	02	--	02	02	--
		Generic Elective Select any one Bucket of courses offered as a major	NCOM304T1	Microprocessor and Interfacing	02	--	02	02	
			NCOM304T2	Data Analytics	02	--	02	02	
			NCOM304T3	Computational Set Theory	02	--	02	02	
	Applied	SEC (Choose any one from	NCOM305P1	Lab based on Microprocessor and Interfacing		02	02		04

		Bucket of major)							
			NCOM305P2	Lab based on Data Analytics		02	02	--	04
			NCOM305P3	Lab based on Computational Set Theory		02	02	---	04
		VSC	NCOM306T	Psychology at Workplace in IT industry	02	-----	02	02	--
	AES,VEC, IKS/ Language Curriculum	AEC-1 English	NCOM-ENG-301	English	02				
		L1-MIL	NCOM-MIL-301	Marathi/Hindi	04				
	VEC/Life Skill Curriculum	JOC-1	NCOM307T	Field Project	02	----	02	02	---
		CC-1	NCOM308T	Cultural Activities /NSS/NCC	02	----	02	02	---

B. Sc. Computer Science Second Year (Semester III) (Level 5.0)
Evaluation/Examination Scheme
[40% Continuous Assessment and 60% of Semester End/ University Assessment]

Year/ Semester and Level		Section	Course Code	Course Name	Credit		Evaluation Method		Total Marks	Max mark	Min Mark
					Theory	Practical	CA	UA			
Second Year Semester III Level 4.5	Major	DSE-5	NCOM301T	Database Management System	03		20	30	50	50	20
			NCOM301P	Lab based on Database Management System	.....	02	--	50	50	50	20
		DSE-6	NCOM302T	Problem Solving using Java Language	03		20	30	50	50	20
			NCOM302P	Java programming Lab	-----	02	---	50	50	50	20
	Supportive	**Minor** Select any one course from Bucket 4	NCOM303T1	Computational Ethics	02	---	20	30	50	50	20
			NCOM303T2	Advance Excel							
			NCOM303T3	Internet Technology							
		Generic Elective Select any	NCOM304T1	Microprocessor and Interfacing	02	---	20	30	50	50	20

		one Bucket 5 of courses offered as a major	NCOM304T2	Data Analytics							
			NCOM304T3	Computational Set Theory							
	Applied	SEC (Choose any one from bucket 6 of major)	NCOM305P1	Lab based on Microprocessor and Interfacing	---	02	---	50	50	50	20
			NCOM305P2	Lab based on Data Analytics							
			NCOM305P3	Lab based on Computational Set Theory							
		VSC	NCOM306T	Psychology at Workplace in IT Industry	02	--	20	30	50	50	20
	AES,VEC, IKS/ Language Curriculum	AEC-1 English	NCOM-ENG-301	English	04	---	40	60	100	100	40
		L1-MIL	NCOM-MIL-301	Marathi/Hindi	04	---	40	60	100	100	40
	VEC/Life Skill Curriculum	JOC-1	NCOM307T	Field Project	02	---	20	30	50	50	20
		CC-3	NCOM308T	Cultural Activities/NSS/NCC	02	---	20	30	50	50	20
							Total Marks		**700**	**700**	**280**

B. Sc. Computer Science Second Year (Semester IV) (Level 5.0)
Teaching Scheme

Year/ Semester and Level		Section	Course Code	Course Name	Credits Assigned			Teaching Scheme (Hrs/ week)	
					Theory	Practical	Total	Theory	Practical
Second Year Semester IV Level 5.0	Major	**DSE-7**	NCOM401T	Compiler Design	03	--	03	03	--
			NCOM401P	Lab based on Compiler Design	...	02	02	--	04
		DSE-8	NCOM402T	OOPs using Python	03	--	03	03	--
			NCOM402P	Python programming Lab	---	02	02	---	04
	Supportive	**Minor-2** Select any one course from Bucket 1	NCOM403T1	Prompt Engineering	02	--	02	02	--
			NCOM403T2	PHP	02	--	02	02	--
			NCOM403T3	Tools for Multimedia	02	--	02	02	--
		Generic Elective /OE-2 Select any one Bucket of courses offered as a major	NCOM404T1	Ethics for Hardware Engineer	02	--	02	02	
			NCOM404T2	Foundation of Machine Learning	02	--	02	02	
			NCOM404T3	Computational Graph Theory	02	--	02	02	

	Applied	SEC (Choose any one from Bucket of major)	NCOM405P1	Lab based on Ethics for Hardware Engineer		02	02		04
			NCOM405P2	Lab based on Foundation of Machine Learning		02	02	--	04
			NCOM405P3	Lab based on Computational Graph Theory		02	02	---	04
		VSC	NCOM406T	Tools for Web Designing	02	-----	02	02	--
	AES,VEC, IKS/ Language Curriculum	AEC-2 English	NCOM-ENG-401	English	04				
		L2-MIL	NCOM-MIL-401	Marathi/Hindi	04				
	VEC/Life Skill Curriculum	VOC-1	NCOM407T	Mindfulness and Wellbeing	02	----	02	02	---
		CC-4	NCOM408T	Fine/Applied/Visual /Performing Arts	02	----	02	02	---

B. Sc. Computer Science Second Year (Semester IV) (Level 5.0)
Evaluation/Examination Scheme
[40% Continuous Assessment and 60% of Semester End/ University Assessment]

Year/ Semester and Level		Section	Course Code	Course Name	Credit		Evaluation Method		Total Marks	Max mark	Min Mark
					Theory	Practical	CA	UA			
Second Year Semester IV Level 5.0	Major	DSE-7	NCOM401T	Compiler Design	03		20	30	50	50	20
			NCOM401P	Lab based on Compiler Design	.....	02	--	50	50	50	20
		DSE-8	NCOM402T	OOPs using Python	03		20	30	50	50	20
			NCOM402P	Python programming Lab	-----	02	---	50	50	50	20
	Supportive	**Minor-2** Select any one course from Bucket 4	NCOM403T1	Prompt Engineering	02	---	20	30	50	50	20
			NCOM403T2	PHP							
			NCOM403T3	Tools for Multimedia							
		Generic Elective /OE-2 Select any one Bucket 5 of courses offered as	NCOM404T1	Ethics for Hardware Engineer	02	---	20	30	50	50	20
			NCOM404T2	Foundation of Machine Learning							
			NCOM404T3	Computational Graph Theory							

		a major									
	Applied	SEC-2 (Choose any one from Bucket 6 of major)	NCOM405P1	Lab based on 8085 Microprocessor	---	02	---	50	50	50	20
			NCOM405P2	Lab based on Data Visualization							
			NCOM405P3	Lab based on Computational Graph Theory							
		VSC	NCOM406T	Tools for Web designing	02	--	20	30	50	50	20
	AES,VEC, IKS/ Language Curriculum	AEC-English	NCOM-ENG-401	English	04	---	40	60	100	100	40
		L2-MIL	NCOM-MIL-401	Marathi/Hindi	02	---	40	60	100	100	40
	VEC/Life Skill Curriculum	VOC-1	NCOM407T	Mindfulness and Wellbeing	02	---	20	30	50	50	20
		CC-4	NCOM408T	Fine/Applied/Visual/Performing Arts	02	---	20	30	50	50	20
								Total Marks	**700**	**700**	**280**

Bucket 4: Minor Subject

* Students will have to choose one subject from Bucket 4 as a Minor subject, from same faculty or discipline or other than DSC (in col. 3)

Semester	Details of Minor Subject	
	Code	Title of Subject
I Semester	NCOM303T1	Computational Ethics
	NCOM303T2	Advance Excel
	NCOM303T3	Internet Technology
II Semester	NCOM403T1	Prompt Engineering
	NCOM403T2	PHP
	NCOM403T3	Tools for Multimedia

Bucket 5: Generic Elective course (GE)

* Students will choose one GE course offered as a generic major

Semester	Details of Generic Elective	
	Code	Title of Subject
III Semester	NCOM304T1	Microprocessor and Interfacing
	NCOM304T2	Data Analytics
	NCOM304T3	Computational Set Theory
IV Semester	NCOM404T1	Ethics for Hardware Engineer
	NCOM404T2	Foundation of Machine Learning
	NCOM404T3	Computational Graph Theory

Bucket 6: Skill Enhancement Course (SEC)

Student should Choose any one from pool of Major

Semester	Details of Skill Enhancement Course (SEC)	
	Code	Title of Subject
III Semester	NCOM305P1	Lab based on Microprocessor and Interfacing
	NCOM305P2	Lab based on Data Analytics
	NCOM305P3	Lab based on Computational Set Theory
IV Semester	NCOM405P1	Lab based on Ethics for Hardware Engineer
	NCOM405P2	Lab based on Foundations of Machine Learning

	NCOM405P3	Lab based on Computational Graph Theory

Curriculum AES,VEC,IKS/Language

Semester	Name of Subject	Details of Skill Enhancement Course (SEC)	
		Code	Title of Subject
III Semester	AEC-3 English	NCOM-ENG-301	English
	MIL –Marathi /Hindi	NCOM-MIL-301	Marathi/ Hindi
	JOC	NCOM307	Field Project
IV Semester	AEC-4 English	NCOM-ENG-402	English
	MIL –Marathi /Hindi	NCOM-MIL-402	Marathi/ Hindi
	VOC	NCOM407	Mindfulness and Wellbeing

Curriculum of Semester –III

Course Code: **NCOM301T**
Section: DSC
Course Title: **Database Management System**
Total Credits: 03
Contact Hours: 45 (Clock Hours)
Marks: 50 **UA:** 30 **CA:** 20
Periods: 45 (45 minutes each)

Objective:

- Understand Database Concepts: To provide students with a clear understanding of basic concepts, principles, and techniques of Database Management Systems (DBMS).
- Database Design: Equip students with the skills required to design a database using data modelling techniques like Entity-Relationship (ER) modelling.
- SQL Proficiency: Develop the ability to write SQL queries for database manipulation, retrieval, and management.
- Normalization: Introduce normalization techniques to optimize database design and reduce redundancy.

Unit-I: Introduction to Databases: (05 periods)
Basic Concepts: Database, DBMS, File System vs. DBMS ,Advantages and Disadvantages of DBMS
Types of DBMS (Hierarchical, Network, Relational, Object-Oriented), Database Users and Architecture: Database Users (Admin, End-users, Application Developers), Three-level architecture: Internal, Conceptual, and External schema, Data Abstraction and Data Independence
Database Models: Entity-Relationship (ER) Model, Attributes, Entities, Relationships, and ER Diagram

Unit-II: Relational Database Model (10 periods)
Relational Model Concepts: Basic Structure of the Relational Model (Tables, Rows, Columns)
Keys: Primary, Foreign, Candidate, Superkey , Integrity Constraints (Domain, Entity, Referential Integrity)
Relational Algebra: Operations: Selection, Projection, Union, Difference, Cartesian Product, Join

Unit-III: Structured Query Language (SQL) (10 periods)
SQL Basics: DDL, DML, DCL commands
SQL Queries: SELECT, INSERT, UPDATE, DELETE
Joins, Subqueries, Aggregate Functions, Grouping

Unit-IV: Database Design and Normalization (15 periods)
Database Design: Steps in Database Design, Conceptual Design and ER to Relational Mapping
Normalization: Functional Dependencies
Normal Forms (1NF, 2NF, 3NF, BCNF), Decomposition and Dependency Preservation
Denormalization: Advantages and Disadvantages of Denormalization, Practical use cases of denormalization

UNIT-V: **Transaction Management and Concurrency Control** (05 periods)
Transactions: Definition, Properties (ACID), Transaction States and Lifecycle
Concurrency Control: Problems of Concurrency (Lost Update, Dirty Read, Inconsistent Data)

Reference Books:

1. Database System Concepts, Abraham Silberschatz, Henry F. Korth, S. Sudarshan, McGraw-Hill, 2010
2. Fundamentals of Database Systems, Ramez Elmasri, Shamkant B. Navathe, Pearson, 2016
3. Database Management Systems, Raghu Ramakrishnan, Johannes Gehrke, McGraw-Hill, 2003

Course Code: **NCOM301P**
Section: DSE(Lab)
Course Title: **Lab course 1 (Lab based on Database Management System)**
Total Credits: 02
Contact Hours: 04 Hours (Week)
Marks: 50 UA: 50

1. Creating and Managing Databases
2. SQL Queries for Data Retrieval
3. Normalization and Denormalization
4. Creating Views and Indexes
5. Stored Procedures and Functions
6. Triggers and Cursors
7. Transaction Management and Concurrency Control
8. Database Backup and Recovery
9. Design and Query a Distributed Database
10. NoSQL Database Implementation

Course Code: **NCOM302T**
Section: DSE
Course Title: **Problem Solving Using Java Programing**
Total Credits: 03
Contact Hours: 45 (Clock Hours)
Marks: 50 (CA: 20 UA: 30)
Periods: 45 (45 minutes each)

Objective:

- To introduce fundamental problem-solving techniques using algorithmic approaches.
- To impart knowledge of Java programming language and its application in solving computational problems.
- To understand the core principles of Object-Oriented Programming (OOP) and apply them in Java.
- To develop skills in writing efficient, modular, and reusable code.
- To explore file handling, error handling, and recursion concepts in Java.
- To prepare students for practical problem-solving through Java programming labs.

Unit-I: Introduction (10 periods)
Problem-solving strategies, algorithm design, and flowcharts. , Overview of Java platform, JDK, JVM, and JRE. ,Java program structure: classes, methods, variables, and data types.
Basic input/output in Java., Control structures (if-else, switch, loops).

Unit-II: Object-Oriented Programming Concepts (05 periods)
Introduction to OOP: classes, objects, encapsulation, inheritance, and polymorphism. , Constructors, method overloading, and method overriding. ,Static members, final classes, and methods.,Abstract classes and interfaces.

Unit-III : Arrays and String Handling (10 periods)
Single and multi-dimensional arrays. ,Array manipulation: sorting and searching algorithms.
Strings in Java: String class, StringBuilder, and StringBuffer.,String manipulation techniques and common operations.

Unit-IV: Exception Handling and File Handling (10 periods)
Introduction to exceptions and error handling.,Try, catch, finally, throw, and throws.,Custom exceptions and best practices for exception handling.,File handling in Java: Reading from and writing to text files using streams.

Unit-V: Recursion, Searching, and Sorting Algorithms (10 periods)
Understanding recursion: principles, base cases, and recursive calls. ,Recursive algorithms: factorial, Fibonacci, Tower of Hanoi. , Searching algorithms: linear search, binary search. ,Sorting algorithms: bubble sort, selection sort, insertion sort, quicksort, merge sort.

Reference Books:

1. Java: The Complete Reference, Herbert Schildt, McGraw-Hill Education, 2021
2. Core Java Volume I – Fundamentals, Cay S. Horstmann, Pearson Education, 2018
3. Head First Java, Kathy Sierra, Bert Bates, O'Reilly Media, 2005
4. Effective Java, Joshua Bloch, Addison-Wesley, 2018

Course Code: NCOM302P
Section: DSE6(Practical)
Course Title: Lab course 2 (Lab based on Java Programming)
Total Credits: 2
Contact Hours: 04 Hours (Week)
Marks: 50 (UA: 50)

Practical based Basic Java Programming

1. Introduction to Java Development Environment: Compiling and executing Java programs.
2. Program to demonstrate control structures (if, switch, loops) in Java.
3. Implementing Object-Oriented Concepts: Classes, objects, and constructors.
4. Practical on arrays and string manipulation in Java.
5. Exception handling: Demonstrating try-catch-finally with file handling.
6. Recursive methods: Factorial, Fibonacci sequence, Tower of Hanoi.
7. Searching and sorting algorithms: Implementing linear search, binary search, and sorting techniques.
8. File handling: Reading and writing files in Java.
9. Introduction to collections: ArrayList, LinkedList, and basic operations.
10. Implementing a mini-project based on learned Java concepts.

Course Code: **NCOM303T1**
Course Title: **Computational Ethics**
Section: Minor
Total Credits: 02
Contact Hours: 30 (Clock Hours)
Marks: 50 (CA: 20 UA: 30)
Periods: 45 (45 minutes each)

Objective:

- Understand the ethical challenges posed by computing technologies.
- Explore the moral responsibilities of computing professionals.
- Analyze real-world case studies related to ethical issues in computing.
- Evaluate the societal impact of emerging technologies like AI, machine learning, and big data.
- Develop ethical reasoning skills to address dilemmas in software development, data privacy, and security.

UNIT-1: Introduction to Ethics and Computing: (10 period)
Definition of ethics and its importance in technology.
History of ethical issues in computing.
Role of computing professionals in promoting ethical practices.
Basic ethical theories: Utilitarianism, Deontology, Virtue Ethics, and their application to computing.

UNIT-2: Professional Codes of Conduct: (10 period)
Overview of professional codes of ethics (ACM Code of Ethics, IEEE Code of Ethics).
Case studies illustrating ethical and unethical behavior in computing.
Responsibility towards users, clients, employers, and society.
Ethical issues in team collaborations and open-source software development.

UNIT-3: Privacy, Data Security, and Surveillance: (10 period)
Ethical challenges in AI and machine learning.
Bias and fairness in algorithms: understanding algorithmic discrimination.
The role of AI in decision-making and its impact on employment and society.
Ethical questions surrounding autonomous systems, including autonomous vehicles and robots.

UNIT-4: Artificial Intelligence and Ethics: (10 period)
Introduction to Internet, Internet Access, Internet Basics, Protocols-TCP/IP,HTTP,FTP, Addressing, World Wide Web(WWW), Web Pages & HTML, Web browsers, Searching for information-search engines. Internet chat. Applications of Internet. Advantages and Disadvantages of Internet

UNIT-5: Intellectual Property, Copyright, and Software Piracy (10 period)
Understanding intellectual property rights (patents, copyrights, trademarks) in the digital world.
Ethical concerns around software piracy and unauthorized access to digital content.
Open-source software vs proprietary software: ethical considerations.
Case studies on intellectual property disputes and their resolutions..

References

1. Ethics for the Information Age, Michael J. Quinn, Pearson, 2019
2. Computers, Ethics, and Society, Deborah G. Johnson, Prentice Hall, 2010
3. The Responsible Software Engineer, Colin Myers, Springer, 2020
4. Computer Ethics, Tom Forester, Perry Morrison, MIT Press, 1994

Course Code: **NCOM303T2**
Course Title: **Advance Excel**
Section: Minor
Total Credits: 02
Contact Hours: 30 (Clock Hours)
Marks: 50 (CA: 20 UA: 30)
Periods: 45 (45 minutes each)

Objective:

- Develop advanced skills in Microsoft Excel for data analysis and management.
- Understand and apply complex formulas and functions for business applications.
- Learn to create and manage PivotTables and PivotCharts for data visualization.
- Master data validation, conditional formatting, and data analysis tools.
- Utilize Excel for statistical analysis and basic modeling.

UNIT-1: Advanced Functions and Formulas: (10 period)
Overview of Excel functions: Logical, Lookup, and Text functions.
Nested functions and array formulas.
Working with date and time functions.
Error handling in formulas (IFERROR, ISERROR).

UNIT-2: Data Management and Analysis:: (05 period)
Data sorting, filtering, and conditional formatting.
Advanced data validation techniques.
Using named ranges for complex formulas.
Introduction to Excel tables: creating, formatting, and referencing.

UNIT-3: PivotTables and PivotCharts: (05 period)
Creating and customizing PivotTables for data analysis.
Using calculated fields and items in PivotTables.
Creating and formatting PivotCharts for data visualization.
Slicers and timelines for filtering PivotTable data.

UNIT-4: Data Visualization and Dashboard Creation: (10 period)
Introduction to data visualization concepts.
Creating advanced charts: combo charts, waterfall charts, and sparklines.
Designing interactive dashboards using Excel features.
Best practices for effective data presentation.

UNIT-5: Statistical Analysis and What-If Analysis (10 period)
Descriptive statistics using Excel functions.
Introduction to data analysis tools: Data Analysis Toolpak.
Performing What-If analysis using Goal Seek and Scenario Manager.
Basic regression analysis and forecasting techniques.

Reference Books:

1. Excel 2019 Power Programming with VBA – Michael Alexander, Dick Kusleika, Wiley
2. Microsoft Excel Data Analysis and Business Modeling – Wayne Winston, Microsoft Press
3. Excel 2016 in Action – Dina M. K. G. P., Manning Publications
4. Excel 2019 for Dummies – Greg Harvey, For Dummies
5. Advanced Excel Reporting for Management – Neale Blackwood, Excel Press

Course Code: **NCOM303T3**
Course Title: **Internet Technology**
Section: Minor
Total Credits: 02
Contact Hours: 30 (Clock Hours)
Marks: 50 (CA: 20 UA: 30)
Periods: 45 (45 minutes each)

Objective:

- Understand the fundamental concepts of the Internet and web technologies.
- Explore protocols and standards that govern Internet communication.
- Gain hands-on experience with web development tools and techniques.
- Learn about client-server architecture and web services.
- Understand security issues and measures related to Internet technologies.

UNIT-1: Introduction to the Internet: (10 period)
Overview of the Internet: History and evolution.
Understanding Internet architecture: LAN, WAN, and Internet.
Key components: Routers, switches, servers, and clients.
Internet service providers (ISPs) and types of Internet connections.

UNIT-2: Internet Protocols and Standards: (05 period)
Overview of the OSI and TCP/IP models.
Understanding IP addressing: IPv4 and IPv6.
Common Internet protocols: HTTP, HTTPS, FTP, SMTP, and DNS.
Introduction to Web standards: W3C and related technologies.

UNIT-3: Web Development Fundamentals: (05 period)
Introduction to HTML, CSS, and JavaScript.
Building web pages: Structure, styling, and interactivity.

Overview of front-end and back-end development. Introduction to responsive web design and frameworks (e.g., Bootstrap)
UNIT-4: Client-Server Architecture and Web Services: (10 period) Understanding client-server architecture. Introduction to web services: REST and SOAP. APIs: Understanding and using application programming interfaces. Data exchange formats: JSON and XML.
UNIT-5: Internet Security and Privacy (10 period) Overview of Internet security threats: Malware, phishing, and DDoS attacks. Principles of secure communication: SSL/TLS and encryption. Best practices for securing web applications and user data. Understanding privacy concerns and regulations (e.g., GDPR).

References

1. Internet Technologies: Principles, Protocols, and Applications – Thomas L. Williams, Pearson
2. HTML and CSS: Design and Build Websites – Jon Duckett, Wiley
3. JavaScript and JQuery: Interactive Front-End Web Development – Jon Duckett, Wiley
4. Web Development and Design Foundations with HTML5 – Terry Felke-Morris, Pearson
5. Computer Networking: A Top-Down Approach – James Kurose, Keith Ross, Pearson

Course Code: **NCOM304T1**
Course Title: **Microprocessor and Interfacing**
Section: Generic/ OE
Total Credits: 02
Contact Hours: 30 (Clock Hours)
Marks: 50 (CA: 20 UA: 30)
Periods: 45 (45 minutes each)

Objective:

- Understand the architecture and organization of microprocessors.
- Develop proficiency in assembly language programming for microprocessors.
- Explore interfacing techniques for input/output devices.
- Learn about memory organization and addressing modes.
- Understand the principles of data conversion and interfacing peripheral devices.

Unit-I: Introduction to Microprocessors: (05 periods)

Overview of microprocessors: Definition, history, and evolution.

Architecture of microprocessors: Data bus, address bus, and control bus.

Organization of microprocessor systems.

Introduction to the 8085 microprocessor: Registers, ALU, and instruction set.

Unit-II: Assembly Language Programming: (10 periods)

Introduction to assembly language programming.

Instruction types: Data transfer, arithmetic, logical, control, and branch instructions.

Writing and debugging simple programs in 8085 assembly language.

Use of assemblers and simulators.

Unit-III: Memory and I/O Interfacing (10 periods)

Memory organization: RAM, ROM, and cache memory.

Addressing modes: Immediate, direct, indirect, and register addressing.

Interfacing input/output devices: Keyboards, displays, and sensors.

Introduction to interfacing techniques: Memory-mapped I/O and I/O-mapped I/O.

Unit-IV: Peripheral Devices and Data Conversion (10 periods)

Overview of peripheral devices: Types and functions.

Interfacing ADC (Analog to Digital Converter) and DAC (Digital to Analog Converter).

Serial and parallel communication: RS-232 and USB protocols.

Understanding timers, counters, and interrupts.

Unit-V: Advanced Microprocessor Concepts:(10 periods)

Introduction to the 8086 microprocessor: Architecture and instruction set.

Segmented memory architecture and addressing modes in 8086.

Introduction to higher-level microprocessors: 80386 and Pentium architectures.

Basics of multiprocessor systems and parallel processing..

Core Reference:

1. Microprocessor Architecture, Programming, and Applications with the 8085 – Ramesh S. Gaonkar, Penram International Publishing, 2013
2. Microprocessors and Interfacing: Programming and Hardware – Douglas V. Hall, Thomson Learning, 2011
3. The Intel Microprocessors – Barry B. Brey, Pearson, 2013

Additional Reference:

1. Microprocessor and Interfacing – N. Senthil Kumar, S. S. Raju, J. A. Thangavel, Wiley, 2016
2. Fundamentals of Microcontrollers and Applications in Embedded Systems – Ramesh S. Gaonkar, Cengage Learning, 2015

Course Code: **NCOM304T2**
Section: Generic/ OE
Course Title: **Data Analytics**
Total Credits: 02
Contact Hours: 30 (Clock Hours)
Marks: 50 (CA: 20 UA: 30)
Periods: 45 (45 minutes each)

Objective:

- Understand the fundamental concepts and importance of data analytics in various fields.
- Develop skills in data collection, cleaning, and preparation.
- Apply statistical methods for analyzing data.
- Utilize data visualization techniques to present insights effectively.
- Learn to use popular data analytics tools and programming languages.
- Understand the ethical considerations in data analytics.

Unit-I: Introduction to Data Analytics: (10 periods)

Overview of data analytics: Definition and significance.

Types of data: Structured, unstructured, and semi-structured data.

Data analytics process: Data collection, preparation, analysis, and presentation.

Tools and technologies in data analytics.

Unit-II: Data Collection and Preparation: (10 periods)

Data sources: Primary vs. secondary data.

Data collection methods: Surveys, experiments, and observational studies.

Data cleaning: Handling missing values, outliers, and inconsistencies.

Data transformation techniques: Normalization, scaling, and encoding.

Unit-III: Statistical Analysis (10 periods)

Descriptive statistics: Measures of central tendency and variability.

Inferential statistics: Hypothesis testing and confidence intervals.

Correlation and regression analysis.

Introduction to probability distributions and their applications.

Unit-IV: Data Visualization (10 periods)

Importance of data visualization in analytics.

Types of visualizations: Charts, graphs, and dashboards.

Tools for data visualization: Tableau, Power BI, and matplotlib (Python).

Best practices for effective data visualization.

Unit-V : Introduction to Data Analytics Tools:(05 periods)

Overview of tools: Excel, R, Python, and SQL.

Using Excel for data analysis: Functions, pivot tables, and charts.

Introduction to R: Data manipulation with dplyr and data visualization with ggplot2.

Basics of Python for data analysis: Libraries such as pandas, NumPy, and matplotlib.

References

1. Data Science for Business – Foster Provost, Tom Fawcett, O'Reilly Media, 2013
2. Python for Data Analysis – Wes McKinney, O'Reilly Media, 2017
3. R for Data Science – Hadley Wickham, Garrett Grolemund, O'Reilly Media, 2017
4. Statistics for Data Science – James D. Miller, Addison-Wesley, 2019

Practical Statistics for Data Scientists – Peter Bruce, Andrew Bruce, O'Reilly Media, 2017

Course Code: **NCOM304T3**
Section: Generic/ OE--1
Course Title: **Computational Set Theory**
Total Credits: 02
Contact Hours: 30 (Clock Hours)
Marks: 50 (CA: 20 UA: 30)
Periods: 45 (45 minutes each)

Objective:

- Understand the fundamental concepts of set theory and its applications in computer science.
- Develop skills in using computational tools to solve problems related to set theory.
- Explore various operations and properties of sets.

- Learn about relations, functions, and their computational representations.
- Understand the concepts of cardinality and infinite sets.

Unit-I: Introduction to Set Theory: (10 periods)

Basic concepts: Sets, elements, subsets, and universal set.

Set notation and representation: Roster and set-builder notation.

Operations on sets: Union, intersection, difference, and complement.

Venn diagrams and their applications.

Unit-II: Relations and Functions: (10 periods)

Definition of relations and properties: Reflexive, symmetric, transitive, and equivalence relations.

Functions: Definition, types, and properties (injective, surjective, bijective).

Composition of relations and functions.

Inverse relations and functions.

Unit-III: Cardinality of Sets (10 periods)

Finite vs. infinite sets.

Countable and uncountable sets.

Cantor's theorem and the concept of cardinality.

Comparing cardinalities of different sets.

Unit-IV: Advanced Set Operations: (10 periods)

Power sets and their properties.

Cartesian products of sets.

Multisets and their operations.

Set theory in databases and information retrieval.

Applications of Set Theory in Computer Science:

Using sets in programming: Data structures (sets, lists, dictionaries). Set operations in SQL and data manipulation. Logic and set theory: Propositions, predicates, and quantifiers. Formal languages and automata theory related to sets.
Unit-V : Computational Tools for Set Theory:(05 periods) Introduction to computational tools (Python, R, etc.) for set operations. Libraries and functions for set manipulations. Case studies and applications of computational set theory in real-world problems.

Reference Books:

1. Set Theory and Its Philosophy – Michael Potter, Oxford University Press, 2004
2. Naive Set Theory – Paul R. Halmos, Springer, 1974
3. Discrete Mathematics and Its Applications – Kenneth H. Rosen, McGraw-Hill, 2011
4. Elements of Set Theory – Herbert B. Enderton, Academic Press, 1977
5. Python for Data Analysis – Wes McKinney, O'Reilly Media, 2017

Course Code: **NCOM305P1** **Section: SEC** Course Title: **Lab on Microprocessor and Interfacing** Total Credits: 02 Contact Hours: 30 (Clock Hours) Marks: 50 (UA: 50) Periods: 45 (45 minutes each)
Objective: To understand the practical knowledge and its implementation of Microprocessor and Interfacing.

1. Writing and executing assembly language programs for 8085.
2. Interfacing and programming ADC and DAC.
3. Simulating I/O devices using microprocessor kits.
4. Conducting experiments with timers and counters.
5. Implementing serial and parallel communication protocols.
6. Hands-on projects involving microprocessor-based systems.

Course Code: NCOM305P2

Section: SEC

Course Title: Lab on Data Analytics

Total Credits: 02

Contact Hours: 30 (Clock Hours)

Marks: 50 (UA: 50)

Periods: 45 (45 minutes each)

Objective: To understand the practical knowledge and its implementation of R Programming

1. Data cleaning and preparation using Excel.
2. Exploratory data analysis with descriptive statistics in R.
3. Visualization of datasets using charts in Excel.
4. Implementing correlation and regression analysis in Python using pandas.
5. Creating basic visualizations with ggplot2 in R.
6. Performing hypothesis testing using sample datasets in Excel.
7. Building a dashboard for data visualization using Tableau.
8. SQL queries for data manipulation and analysis on a sample database.
9. Data analysis project: Collecting, cleaning, and analyzing a dataset.
10. Ethical case study discussion: Analyzing the ethical implications of a data analytics project.

Course Code: NCOM305P3

Section: SEC

Course Title: Lab on Computational Set Theory

Total Credits: 02

Contact Hours: 30 (Clock Hours)

Marks: 50 (UA: 50)

Periods: 45 (45 minutes each)

Objective: To understand the practical knowledge and its implementation Computational Set Theory

1. Implement basic set operations (union, intersection, difference) using Python.
2. Create Venn diagrams for visualizing set operations.
3. Write programs to find the power set of a given set.
4. Implement functions to check properties of relations (reflexive, symmetric, transitive).
5. Perform operations on multisets and analyze their properties.
6. Use SQL to demonstrate set operations on a sample database.
7. Explore functions and relations through programming exercises.
8. Conduct experiments to compare the cardinality of finite and infinite sets.
9. Develop a program to simulate Cartesian products of sets.
10. Case study: Analyze a real-world problem using set theory and present findings.

Course Code: **NCOM306T**
Section: VSC
Course Title: **Psychology at Workplace in IT Industry**
Total Credits: 02
Contact Hours: 30 (Clock Hours)
Marks: 50 (CA: 20 UA: 30)
Periods: 45 (45 minutes each)

Objective:

- Understand fundamental psychological concepts relevant to the workplace.
- Explore the impact of psychological factors on employee behavior and performance in the IT industry.
- Analyze the role of motivation, leadership, and team dynamics in organizational settings.
- Develop skills for managing stress and promoting mental well-being in the workplace.
- Learn to apply psychological principles to improve workplace communication and collaboration.

Unit-I: : Introduction to Industrial Psychology (10 periods)

Overview of industrial psychology: Definition and significance.

Historical development and key theories in industrial psychology.

Research methods in industrial psychology: Surveys, experiments, and observational studies.

Unit-II: Employee Motivation and Job Satisfaction (10 periods)

Theories of motivation: Maslow's hierarchy of needs, Herzberg's two-factor theory, and expectancy theory.

Factors influencing job satisfaction and performance.

Techniques for enhancing motivation and job satisfaction in IT professionals.

Unit-III: Leadership and Team Dynamics (05 periods)

Leadership styles and their impact on employee performance.

The role of effective communication in teams.

Team dynamics: Stages of team development, roles, and conflict resolution.
Unit-IV: Stress Management and Mental Health (10 periods) Understanding workplace stress: Causes, consequences, and coping strategies. Techniques for managing stress in the IT industry. Promoting mental health and well-being in the workplace. **Organizational Culture and Change** The importance of organizational culture in the IT industry. Strategies for managing organizational change. The role of psychology in organizational development and transformation.
Unit-V: Ethical Considerations in Industrial Psychology (10 periods) Understanding ethics in workplace psychology. Issues of privacy, consent, and confidentiality. The role of psychologists in the workplace: Responsibilities and ethical dilemmas.

Reference books:

1. Work Psychology: Understanding Human Behaviour in the Workplace – John Arnold, et al., Routledge, 2016
2. The Psychology of Workplace Motivation – Andrew J. DuBrin, South-Western Cengage Learning, 2015
3. Managing Human Behavior in Public and Nonprofit Organizations – Robert B. Denhardt, et al., Cengage Learning, 2013
4. Psychology and Work Today – Duane Schultz, Sydney Ellen Schultz, Pearson, 2016

Curriculum of Semester –IV

Course Code: **NCOM401T**
Section: DSC
Course Title: **Compiler Design**
Total Credits: 03
Contact Hours: 45 (Clock Hours)
Marks: 50 **UA**: 30 **CA**: 20
Periods: 45 (45 minutes each)

Objective:

- To understand the fundamental concepts of compilers, their architecture, and phases.
- To learn lexical analysis and syntax analysis techniques.
- To explore parsing techniques, semantic analysis, and code generation.
- To study optimization techniques in code generation.
- To apply knowledge of automata theory and formal languages in compiler design.

Unit-I: Introduction to Compilers: (05 periods)

Compilers and Translators: Overview

Phases of a Compiler

Structure of a Compiler

Compiler construction tools

Cross Compiler, Just-in-Time Compiler

Unit-II: Lexical Analysis (10 periods)

Role of Lexical Analyzer

Input Buffering

Regular Expressions

Finite Automata and its types

Lexical Errors

Unit-III: Syntax Analysis (10 periods)

Stacks- Introduction & Definition, Application of Stack, Various Representation of Stack, Operation on stack (Push and Pop) Hierarchy of Operation, Representation of Arithmetic Expression (Infix, Postfix, Prefix) Multiple Stack. Evaluation of postfix expressions and their conversions

Unit-IV: Semantic Analysis and Intermediate Code Generation (05 periods) Syntax-directed Definitions Type Checking and Type Systems Symbol Table Management Intermediate Code Representation: Three Address Code, Quadruples, Triples Boolean Expressions
UNIT-5: Code Optimization (05 periods) Optimization Techniques: Peephole Optimization Local and Global Optimization Basic Blocks and Flow Graphs Loop Optimization Data Flow Analysis

Reference Books:

1. Aho, A.V., Sethi, R., & Ullman, J.D., Compilers: Principles, Techniques, and Tools, Addison Wesley.
2. Appel, A.W., Modern Compiler Implementation in C, Cambridge University Press.

Course Code: **NCOM401P** **Section: DSE(Lab)** Course Title: **Lab course (Lab based on Compiler Design)** Total Credits: 02 Contact Hours: 04 Hours (Week) Marks: 50 UA: 50
1. Lexical Analyzer Construction: Develop a program to recognize tokens and identifiers in a given source program. 2. Regular Expression to NFA Conversion: Implement a tool to convert a regular expression into a Non-deterministic Finite Automaton (NFA). 3. Syntax Analyzer for Arithmetic Expressions: Design and implement a parser for arithmetic expressions using recursive descent parsing. 4. LL(1) Parser Implementation: Construct a predictive parser for a given grammar. 5. SLR Parser: Implement a Simple LR (SLR) parser for a given grammar. 6. Syntax Tree Construction: Create a syntax tree for a given expression or statement.

7. Intermediate Code Generation: Write a program to generate intermediate code for simple mathematical expressions.
8. Type Checking: Develop a tool to perform type checking for expressions in a small programming language.
9. Optimization Techniques: Implement peephole optimization for intermediate code.
10. Code Generation for Expressions: Generate target code for a given set of intermediate instructions.

Course Code: **NCOM402T**
Section: DSC2
Course Title: **OOPs using Python**
Total Credits: 03
Contact Hours: 45 (Clock Hours)
Marks: 50 **UA:** 30 **CA:** 20
Periods: 45 (45 minutes each)

Objective:

- To introduce the fundamental concepts of object-oriented programming (OOP) using Python.
- To understand the principles of classes, objects, inheritance, and polymorphism.
- To develop problem-solving skills through the application of OOP techniques.
- To design Python programs using object-oriented concepts for real-world problems.
- To gain hands-on experience with Python for object-oriented programming.

Unit-I: Introduction to Object-Oriented Programming: (05 periods)

Overview of OOP Concepts

Need for OOP: Procedural vs. Object-Oriented Programming

Key OOP Concepts: Class, Object, Abstraction, Encapsulation, Polymorphism

Introduction to Python as an Object-Oriented Language

Python Basics: Variables, Data Types, Control Flow

Unit-II: Classes and Objects (10 periods) Defining Classes and Objects in Python Constructors and Destructors Instance Variables and Methods Class Variables and Methods Static Methods Self-Parameter in Python
Unit-III: Inheritance: (10 periods) Concept of Inheritance in OOP Types of Inheritance: Single, Multiple, Multilevel, Hierarchical Overriding Methods Super() Function and Method Resolution Order (MRO) Composition vs. Inheritance
Unit-IV: Polymorphism and Encapsulation (15 periods) Method Overloading and Method Overriding Operator Overloading Encapsulation and Data Hiding Access Modifiers: Private, Public, Protected Property Functions: Getters and Setters
UNIT-5: Exception Handling and File Handling: (05 periods) Exception Handling in Python: Try, Except, Finally Custom Exceptions Importance of Exception Handling in OOP File Handling in Python: Reading and Writing Files Using OOP Concepts in File Management

Reference Books:

1. Guttag, J.V., Introduction to Computation and Programming Using Python, MIT Press.
2. Liang, Y.D., Introduction to Programming Using Python, Pearson.

Course Code: **NCOM402P**
Section: DSE1(Lab)
Course Title: **Lab course 5 (Python Programming Lab)**
Total Credits: 02
Contact Hours: 04 Hours (Week)
Marks: 50 UA: 50

1. Class and Object Implementation: Create a Python program to define a class, instantiate objects, and demonstrate class methods.
2. Constructor and Destructor: Write a program to demonstrate the use of constructors and destructors in Python.
3. Single and Multiple Inheritance: Implement examples of single and multiple inheritance in Python.
4. Polymorphism: Develop a Python program to demonstrate method overriding and operator overloading.
5. Encapsulation and Data Hiding: Write a Python program to illustrate the concepts of encapsulation and access modifiers.
6. File Handling with OOP: Implement a Python program that uses classes for reading and writing files.
7. Exception Handling: Develop a Python program that includes custom exceptions and demonstrates exception handling in OOP.
8. Abstract Classes and Interfaces: Create a Python program using abstract classes and interfaces.
9. Iterators and Generators: Implement a Python program to show the use of iterators and generators with OOP principles.
10. Design Patterns in Python: Write a Python program to implement a basic design pattern such as Singleton or Factory.

Course Code: **NCOM403T1**
Course Title: **Prompt Engineering**
Section: Minor 4
Total Credits: 02
Contact Hours: 30 (Clock Hours)
Marks: 50 (CA: 20 UA: 30)
Periods: 45 (45 minutes each)

Objective:

- To introduce the concept of prompt engineering and its role in leveraging AI models.
- To understand the principles of creating effective prompts for language models like GPT.
- To develop the ability to design prompts for various natural language processing (NLP) tasks.
- To explore advanced techniques for improving prompt performance and results.
- To gain hands-on experience in using prompts to solve real-world problems using AI language models.

UNIT-1: Introduction to Prompt Engineering: (10 period)
Definition and Importance of Prompt Engineering
Overview of Large Language Models (LLMs) and Generative AI
Types of Prompts: Instruction, Demonstration, Contextual Prompts
Introduction to OpenAI GPT and Other AI Models
Basic Prompt Design and Structure

UNIT-2: Designing Prompts for NLP Tasks: (10 period)
Text Generation: Crafting Effective Prompts for Creative Writing
Summarization: Creating Prompts for Summarizing Long Texts
Translation: Designing Prompts for Multilingual Models
Question-Answering: Generating Prompts for Factual Retrieval
Sentiment Analysis: Using Prompts to Classify Emotions in Text

UNIT-3: Few-shot and Zero-shot Learning: (10 period)
Zero-shot Prompting: Using Models Without Task-specific Data
Few-shot Prompting: Enhancing Model Performance with Minimal Examples
Comparison of Few-shot, Zero-shot, and Fine-tuning Approaches
Best Practices for Selecting Few-shot Examples
Challenges and Limitations of Zero-shot and Few-shot Learning

UNIT-4: Advanced Prompt Techniques (10 period)
Chain of Thought (CoT) Prompts
Role-based and Instruction-based Prompts
Dynamic Prompting: Adjusting Prompts Based on Model Feedback
Interactive and Multi-turn Prompting

Prompt Tuning and Optimization Strategies
UNIT-5: Evaluation and Refinement of Prompts (05 period) Metrics for Evaluating Prompt Effectiveness: Fluency, Relevance, Coherence Debugging and Refining Prompts for Optimal Output Common Errors in Prompt Engineering and Their Fixes Ethical Considerations in Prompt Engineering: Bias, Misuse, and Safety Best Practices for Safe and Responsible AI Use

Reference Books

1. Jurafsky, D., & Martin, J.H., Speech and Language Processing, Pearson.
2. Russell, S.J., & Norvig, P., Artificial Intelligence: A Modern Approach, Pearson.

Course Code: **NCOM403T2** Course Title: **PHP** **Section: Minor 4** Total Credits: 02 Contact Hours: 30 (Clock Hours) Marks: 50 (CA: 20 UA: 30) Periods: 45 (45 minutes each)
Objective: • To introduce students to server-side web programming using PHP. • To teach the fundamentals of PHP scripting for dynamic web development. • To equip students with knowledge of integrating PHP with databases like MySQL. • To develop the ability to create interactive and data-driven web applications. • To enhance understanding of session management, security, and web services in PHP.
UNIT-1: Introduction to PHP: (10 period) Basics of Web Development Introduction to Server-Side Programming Overview of PHP: History, Features, and Installation PHP Syntax, Variables, Constants Data Types, Operators, and Control Structures in PHP

UNIT-2: Working with Forms and User Input: (10 period) PHP and HTML Integration Handling Form Data with GET and POST Validating Form Data Sanitizing User Input Redirecting Users and Handling Errors
UNIT-3: Arrays, Strings, and Functions in PHP:: (10 period) Working with Arrays: Indexed, Associative, and Multidimensional String Functions and Manipulation Creating and Using Functions in PHP Scope of Variables: Local and Global Passing Arguments by Reference and Value
UNIT-4: File Handling and Sessions (10 period) File Handling: Reading and Writing Files File Uploads in PHP PHP Sessions: Creating and Managing Sessions Cookies: Setting, Reading, and Deleting Maintaining State and Persistent Data in Web Applications
UNIT-5: Database Connectivity with MySQL: (05 period) Introduction to MySQL Database Connecting PHP to MySQL Using MySQLi and PDO Performing CRUD Operations: Create, Read, Update, Delete Using PHP to Execute SQL Queries Error Handling in Database Operations

Reference Books:

1. Luke Welling, Laura Thomson, PHP and MySQL Web Development, Pearson.
2. Brett McLaughlin, PHP & MySQL: The Missing Manual, O'Reilly Media.

Course Code: **NCOM403T3**
Course Title: **Tools for Multimedia**
Section: Minor 4
Total Credits: 02
Contact Hours: 30 (Clock Hours)
Marks: 50 (CA: 20 UA: 30)
Periods: 45 (45 minutes each)

Objective:

- To provide an understanding of various multimedia tools and their applications.
- To equip students with skills in multimedia content creation and editing using modern tools.
- To explore techniques for working with text, audio, video, images, and animation.
- To gain hands-on experience with multimedia software for developing interactive projects.
- To understand the integration of multimedia elements into web and application development.

UNIT-1: Introduction to Multimedia Tools: (10 period)

Introduction to Multimedia: Concepts and Types (Text, Audio, Video, Animation)
Multimedia Systems and Applications
Overview of Popular Multimedia Tools (Adobe Photoshop, Audacity, Premiere Pro, etc.)
Formats and Standards in Multimedia (Image formats: JPEG, PNG; Video formats: MP4, AVI; Audio formats: MP3, WAV)

UNIT-2: Image Editing Tools: (10 period)

Basics of Digital Image Processing
Working with Adobe Photoshop or GIMP: Image Manipulation and Enhancement
Layering and Compositing Techniques
Image Formats and Exporting Options
Color Correction, Filters, and Effects

UNIT-3: Audio Editing Tools:: (10 period)

Basics of Digital Audio Processing
Introduction to Audacity or Adobe Audition
Recording, Importing, and Editing Audio Files
Sound Effects, Filters, and Equalization
Audio Formats and Compression

UNIT-4: Video Editing Tools (10 period)

Introduction to Video Formats and Compression
Basics of Video Editing with Adobe Premiere Pro or Final Cut Pro
Trimming, Cutting, and Transitions in Videos
Adding Text, Titles, and Effects to Videos
Rendering and Exporting Videos for Different Platforms

UNIT-5: Animation and Graphics Tools: (05 period)

Basics of 2D Animation using Adobe Animate or Blender
Timeline and Frame-by-Frame Animation Techniques
Adding Audio and Interactivity to Animation
Basics of 3D Modeling and Animation (Blender)
Rendering and Exporting Animations.

Reference Books:

1. Tay Vaughan, Multimedia: Making it Work, McGraw Hill.

2. Ze-Nian Li, & Mark S. Drew, Fundamentals of Multimedia, Pearson.

Course Code: **NCOM404T1** Course Title: **Ethics for Hardware Engineer** **Section: Generic Elective/ O. E-4** Total Credits: 02 Contact Hours: 30 (Clock Hours) Marks: 50 (CA: 20 UA: 30) Periods: 45 (45 minutes each)
Objective: • To introduce the ethical considerations specific to hardware engineering. • To develop an understanding of professional responsibilities and moral challenges faced by hardware engineers. • To explore issues related to safety, privacy, intellectual property, and environmental impact in hardware design and manufacturing. • To foster critical thinking about ethical decision-making in hardware engineering projects. • To analyze real-world case studies and apply ethical theories to engineering practices.
UNIT-1: Introduction to Ethics in Engineering (10 period) Definition and importance of ethics Ethical frameworks: Consequentialism, Deontology, Virtue Ethics Ethics in engineering professions
UNIT-2: Professional Responsibility and Codes of Conduct (10 period) Codes of ethics from IEEE, ACM, and other professional bodies Responsibility to clients, employers, and society Ethical leadership in hardware engineering
UNIT-3: Safety and Risk in Hardware Engineering (10 period) Importance of safety in hardware design Risk assessment and mitigation strategies Case studies on hardware-related failures and ethical consequences
UNIT-4: Privacy, Security, and Ethical Challenges (10 period) Ethical issues in hardware that impact privacy and security Encryption, hacking, and hardware vulnerabilities Case studies: Privacy breaches and hardware-related security flaws

UNIT-5: Environmental Ethics in Hardware Engineering (05 period) Environmental impact of hardware production and disposal Sustainable design and e-waste management Ethical issues in the lifecycle of hardware products

Reference Book

1. Ethics in Engineering by Mike W. Martin and Roland Schinzinger
2. Engineering Ethics by Charles E. Harris, Michael S. Pritchard, and Michael J. Rabins
3. Environmental Ethics for Engineers by Alastair S. Gunn
4. The Ethical Engineer: An "Ethics Construction Kit" for Technical Professionals by Robert McGinn

Course Code: **NCOM404T2** Course Title: **Foundation of Machine Learning** **Section: Generic Elective/ O. E-2** Total Credits: 02 Contact Hours: 30 (Clock Hours) Marks: 50 (CA: 20 UA: 30) Periods: 45 (45 minutes each)
Objective: The primary objective of this course is • To introduce fundamental concepts and techniques in machine learning. • To understand various supervised and unsupervised learning algorithms. • To implement and apply machine learning models to real-world problems. • To develop skills in data preprocessing, model evaluation, and performance improvement. • To explore ethical considerations and limitations of machine learning models.
UNIT-1: Introduction to Machine Learning (10 period) What is machine learning? Types of machine learning: Supervised, Unsupervised, Reinforcement Applications of machine learning in various domains
UNIT-2: Data Preprocessing and Feature Engineering (10 period) Data cleaning, normalization, and transformation Feature extraction and selection Handling missing data and outliers

UNIT-3: Supervised Learning Algorithms (10 period) Linear Regression Logistic Regression Decision Trees Support Vector Machines (SVMs) k-Nearest Neighbors (k-NN)
UNIT-4: Unsupervised Learning Algorithms (05 period) Clustering: k-Means, Hierarchical Clustering Dimensionality Reduction: PCA, LDA Anomaly Detection
UNIT-4: Evaluation Metrics and Model Validation (05 period) Train-Test Split, Cross-Validation Confusion Matrix, Precision, Recall, F1 Score ROC-AUC Curve Bias-Variance Tradeoff

Reference Book

1. Pattern Recognition and Machine Learning by Christopher Bishop
2. Machine Learning: A Probabilistic Perspective by Kevin P. Murphy
3. Hands-On Machine Learning with Scikit-Learn and TensorFlow by Aurélien Géron
4. Deep Learning by Ian Goodfellow, Yoshua Bengio, and Aaron Courville

Course Code: **NCOM404T3** Course Title: **Computational Graph Theory** **Section: Generic Elective/ O. E-2** Total Credits: 02 Contact Hours: 30 (Clock Hours) Marks: 50 (CA: 20 UA: 30) Periods: 45 (45 minutes each)
Objective: Students successfully completing this course should be able to: • To understand the basic concepts and structures of graph theory. • To learn computational methods for solving graph-related problems. • To explore applications of graph theory in computer science and other disciplines. • To develop proficiency in implementing graph algorithms for various real-world

problems. • To analyze and evaluate the efficiency of algorithms in graph theory.
UNIT-1: Introduction to Graph Theory (10 period) Definition of graphs: Directed, Undirected Graph terminology: Vertices, Edges, Degree, Paths, Cycles Types of graphs: Simple, Complete, Bipartite, Planar, Trees Real-world applications of graphs
UNIT-2: Graph Representations (10 period) Adjacency matrix Adjacency list Incidence matrix Efficiency of graph representations
UNIT-3: Graph Traversal Algorithms (10 period) Breadth-First Search (BFS) Depth-First Search (DFS) Applications of BFS and DFS in problem-solving
UNIT-4: Shortest Path Algorithms (05 period) Dijkstra's algorithm Bellman-Ford algorithm Floyd-Warshall algorithm Applications in networking and transport
UNIT-5: Minimum Spanning Tree (MST) Algorithms (05 period) Kruskal's algorithm Prim's algorithm Applications of MST in network design and optimization

Core References:

1. Introduction to Graph Theory by Douglas B. West
2. Graph Theory with Applications by Bondy and Murty
3. Algorithm Design Manual by Steven S. Skiena
4. Graph Algorithms in the Language of Linear Algebra by Jeremy Kepner and John Gilbert

Course Code: **NCOM405P1**
Section: SEC
Course Title: **Lab course 5 (Lab based on Ethics of Hardware Engineer)**
Total Credits: 02
Contact Hours: 04 Hours (Week)
Marks: 50 UA: 50

1. Ethical Case Study Analysis
   Choose a real-life hardware engineering failure and analyze the ethical issues involved. Present findings in a group discussion.
2. Safety Risk Assessment
   Conduct a safety risk assessment on a hardware project and develop a mitigation plan.
3. Environmental Impact Analysis
   Analyze the lifecycle of a hardware component (e.g., a microprocessor) and assess its environmental footprint.
4. Privacy and Security Breach Simulation
   Simulate a hardware security breach scenario and propose ethical solutions to mitigate the risks.
5. Intellectual Property Rights Analysis
   Review a case of hardware patent infringement and analyze the ethical considerations.
6. Design for Sustainability
   Develop a hardware design with sustainability in mind and present an environmental impact report.
7. E-Waste Management Project
   Propose an ethical solution to the issue of e-waste management in your community or institution.
8. Ethical Leadership Roleplay
   Role-play a scenario where you are a hardware engineer facing an ethical dilemma at your workplace.
9. Whistleblowing and Professional Integrity
   Explore the concept of whistleblowing by analyzing a famous whistleblower case in the tech industry.
10. Hardware Vulnerabilities Testing
    Investigate vulnerabilities in a hardware system and provide ethical recommendations for improvements.

Course Code: **NCOM405P2**
Section: SEC
Course Title: **Lab course 5 (Lab based on Foundations of Machine Learning)**
Total Credits: 02
Contact Hours: 04 Hours (Week)
Marks: 50 UA: 50

1. Data Preprocessing and Visualization
   Preprocess a real-world dataset, handle missing values, and visualize key patterns using data visualization techniques.
2. Linear Regression Implementation
   Build and train a linear regression model to predict housing prices using a dataset like the Boston Housing dataset.
3. Classification with Logistic Regression
   Implement logistic regression on a dataset (e.g., Breast Cancer dataset) to classify binary outcomes and evaluate the model's performance.
4. Decision Trees for Classification
   Use decision tree algorithms to classify data from the Iris dataset and visualize the decision tree structure.
5. Clustering with k-Means
   Apply the k-Means clustering algorithm to a dataset (e.g., customer segmentation) and evaluate the clusters formed.
6. Principal Component Analysis (PCA)
   Reduce the dimensionality of a dataset using PCA and visualize the impact of dimensionality reduction on model performance.
7. Random Forest for Classification
   Implement Random Forest on a real-world classification dataset (e.g., Titanic survival dataset) and evaluate its performance using accuracy, precision, and recall.
8. Neural Network Implementation
   Design and train a simple neural network using TensorFlow or Keras to recognize handwritten digits from the MNIST dataset.
9. Model Evaluation using Cross-Validation
Apply k-fold cross-validation to evaluate the performance of a machine learning model (e.g., SVM on a classification task).
10. Ethical Analysis of a Machine Learning Model
Analyze the ethical implications of a machine learning model by exploring issues related to bias, fairness, and transparency in a given dataset.

Course Code: **NCOM405P3**
Section: SEC
Course Title: **Lab course (Lab based on Computational Graph Theory)**
Total Credits: 02
Contact Hours: 04 Hours (Week)
Marks: 50 UA: 50

1. **Graph Representation and Visualization**
   Implement various graph representations (adjacency matrix, adjacency list) for a given graph and visualize using graph plotting libraries (e.g., NetworkX in Python).
2. **Breadth-First Search (BFS) and Depth-First Search (DFS)**
   Write programs to perform BFS and DFS on a graph and explore their applications (e.g., finding connected components or detecting cycles).
3. **Shortest Path Implementation**
   Implement Dijkstra's and Bellman-Ford algorithms to find the shortest paths in a weighted graph. Compare their performance.
4. **Minimum Spanning Tree Construction**
   Implement Kruskal's and Prim's algorithms to find the MST of a given graph. Apply them to a network design problem.
5. **Graph Coloring Algorithm**
   Develop a program to color a graph with the minimum number of colors (Chromatic number). Apply it to a scheduling problem.
6. **Maximum Matching in Bipartite Graphs**
   Implement the Hungarian algorithm for finding the maximum matching in a bipartite graph and explore its applications in job assignment problems.
7. **Network Flow and Ford-Fulkerson Algorithm**
   Implement the Ford-Fulkerson algorithm to solve the maximum flow problem in a flow network and apply it to a transportation or communication problem.
8. **Planar Graph Testing**
   Implement algorithms to test whether a graph is planar or not. Visualize planar embeddings of graphs.
9. **Eulerian and Hamiltonian Circuits**
   Develop programs to find Eulerian circuits and Hamiltonian paths in given graphs. Explore real-world applications such as the traveling salesman problem (TSP).
10. **Graph Isomorphism**
    Implement algorithms to check for graph isomorphism and apply them to molecular structures or chemical compound analysis.

Course Code: **NCOM406T**
Course Title: **Tools for Web Designing**
Section: VSC-4
Total Credits: 02
Contact Hours: 30 (Clock Hours)
Marks: 50 (CA: 20 UA: 30)
Periods: 45 (45 minutes each)

Objective: The primary objective of this course is

- To introduce the fundamentals of web design and development.
- To understand the use of various tools for creating and maintaining websites.
- To explore responsive web design techniques and frameworks.
- To learn about content management systems (CMS) and website hosting.
- To gain practical experience with web design tools and software.

UNIT-1: Introduction to Web Design (10 period)
Basics of web design: Structure, Layout, Aesthetics
Introduction to HTML, CSS, and JavaScript
Web design principles: Usability, Accessibility, and SEO

UNIT-2: HTML and CSS Fundamentals (10 period)
Structure of an HTML document
HTML5 elements and attributes
CSS basics: Selectors, properties, and layouts
Responsive web design using media queries
Introduction to CSS Flexbox and Grid systems

UNIT-3: Introduction to Web Design Tools (10 period)
Text editors: Visual Studio Code, Sublime Text
Graphic design tools: Adobe Photoshop, Figma, Canva
Website builders: WordPress, Wix, Squarespace
Browser developer tools for debugging and testing

UNIT-4: Responsive Web Design (10 period)
Responsive vs. Adaptive design
Introduction to Bootstrap and other CSS frameworks
Creating fluid layouts and responsive navigation
Designing for different screen sizes (mobile, tablet, desktop)

UNIT-5: JavaScript for Web Development (05 period)
Introduction to JavaScript basics: Variables, functions, events
DOM manipulation using JavaScript
Introduction to JavaScript libraries: jQuery
Basic form validation using JavaScript

References Book

1. HTML & CSS: Design and Build Websites by Jon Duckett
2. JavaScript and JQuery: Interactive Front-End Web Development by Jon Duckett
3. Learning Web Design: A Beginner's Guide to HTML, CSS, JavaScript, and Web Graphics by Jennifer Robbins
4. Responsive Web Design with HTML5 and CSS by Ben Frain

Course Code: **NCOM407T**
Course Title: **Mindfulness and Wellbeing**
Section: VOC-4
Total Credits: 02
Contact Hours: 30 (Clock Hours)
Marks: 50 (CA: 20 UA: 30)
Periods: 45 (45 minutes each)

Objective: The primary objective of this course is

- To introduce the principles and practices of mindfulness.
- To explore the role of mindfulness in enhancing mental, emotional, and physical well-being.
- To develop practical mindfulness techniques for managing stress and improving focus.
- To understand the scientific basis and benefits of mindfulness in various aspects of life.
- To encourage self-awareness, emotional regulation, and overall well-being.

UNIT-1: Introduction to Mindfulness and Wellbeing (10 period)
Definition and history of mindfulness
Concept of well-being: Physical, emotional, and psychological
Relationship between mindfulness and well-being
Overview of mindfulness traditions: Buddhist, secular, modern approaches

UNIT-2: The Science of Mindfulness (10 period)
Neuroscience of mindfulness
Psychological and physiological effects of mindfulness on the brain and body
Mindfulness and emotional regulation
Mindfulness-based interventions: Mindfulness-Based Stress Reduction (MBSR), Mindfulness-Based Cognitive Therapy (MBCT)

UNIT-3: Basic mindfulness meditation: Breathing exercises, body scan Focus and attention training Loving-kindness meditation Walking meditation, mindful movement, and yoga practices
UNIT-4: Mindfulness for Stress Reduction (10 period) Stress: Causes, symptoms, and impacts on health Mindfulness as a tool for managing stress and anxiety Practicing mindfulness in daily life: Mindful eating, communication, and routines Coping with challenges through mindfulness
UNIT-5: Mindfulness in Personal Development and Wellbeing (05 period) Self-awareness and self-compassion Building emotional resilience Enhancing creativity and problem-solving through mindfulness Mindfulness for improving interpersonal relationships

Reference Books

1. Wherever You Go, There You Are by Jon Kabat-Zinn
2. The Miracle of Mindfulness by Thich Nhat Hanh
3. Radical Acceptance: Embracing Your Life with the Heart of a Buddha by Tara Brach
4. The Headspace Guide to Meditation and Mindfulness by Andy Puddicombe