LOK JAGRUTI KENDRA UNIVERSITY

Syllabus for two Years

School of Computer Applications,

Master of Computer Applications (MCA)

Semester - I

Course Code	240110101			
Category	Core Subject			
Course Title	Programming in C (C)			
Scheme and Credits	Theory	Tutorial	Lab	Credits
	0	1	6	4
Pre-requisites (if any)	No			

1. **Course Objectives:**

1	To recall facts, concepts, and basic syntax in C programming.
2	To comprehend the meaning of code snippets, algorithms, and data structures in C.
3	To use knowledge and skills to solve problems and write functional code in C.
4	To break down code, identify errors, and understand the flow of control in programs.
5	To judge the quality, efficiency, and effectiveness of a C program using various testing techniques.

2. **Course contents:**

Module	Content	Weightage
Unit I	**Introduction to programming in C:** What is programming, Introduction to program and programming languages, Simple Program in C, compiler, interpreter, loader, linker, C program execution, Classification of Programming Languages, Flow Chart & Algorithm **Good programming practices**: Comments, Meaningful identifiers name, Escape sequence, Structure of C Program, Tokens, Concept of Identifier, Variable, Constant and Keywords, Data types in C. **Basics of C:** Program statements, declarations, How the computer stores data in memory, Operators and Expressions, Type casting & type conversion.	20%
Unit II	**Control Statements, Arrays & Strings:** **Control Statements:**	20%

	Specifying test condition for selection and iteration, writing test expression,Conditional execution and selection, Iteration and repetitive Execution: goto statement, for, while, do-while loops; variations of for loop, break, continue statement, nested loops. **Arrays & strings:** One-dimensional Array, Strings, String: One dimensional Array, Multi-dimensional array, Array of string. Writing first program with the help of flow chart and user define function.	
Unit III	**Functions and Pointers:** **Function-**What is function, need of function, types of function, Passing array to Function, Scope and Extent, Storage class **Recursion-**Concept of recursion, Types of recursion, comparison of iteration and recursion **Pointer** - Concept of memory addresses, use of the (&) operator. **Pointer Execution** – Declaring and Initializing a pointer, Indirection Operator and dereferencing a pointer. **Pointer Arithmetic** – Assignment, Addition-subtraction with integers, pointer comparison. **Use of Pointers** – Returning more than one value from the function (Call by value and call by References concept). **Pointers and Arrays** – One dimensional Arrays and pointers, Difference between array name and pointers, passing one dimensional and two dimensional array to functions, Pointers for character array(strings), Array of pointers 2-D Array and pointer, pointer to function. **Dynamic memory allocation** – Array allocation, Memory freeing, Memoryreallocation. **Informative**- void Pointers, Null pointers, Pointers to pointers, Memory leak and Memory corruption, pointer to constant and constant pointer.	20%
Unit IV	**User Defined Data types:** **User Defined Data types – Basic structure operations-** Declaring structures and structure variables, Accessing the members of the structure, Initialization of structure, Structure comparisons and copying, typedef and its use. **Advanced** - Nested structures, Array within the structure and Array of structures,Structures with functions and structures with pointers. Union and its operations Enumerated data type, bit field	20%

	(fseek(),ftell(),rewind(),fgetpos(),fsetpos()) **Error handling during File operations** **Files of records** – Working with files of records (fread() and fwrite()). **Other functions** – Renaming a file, Removing a file, Command line Arguments	
Unit V	**Files in C and Pre-processor directives:** **Files in C: Introduction – Streams in C, Types of Files.** **Files concept in C** – Declaration of File pointer, opening a file, closing a file various modes to open the file, Handling file functions for reading data from files – fscanf (), fgetw (), fgets (), fgetc (), fread(), Handling file functions for writing data into files- **f**printf, fputw, fputs(), fputc(), fwrite(),Detecting the end of the file – what is EOF and feof() function **Types of the files** – Binary and Text files. **Types of access to the file** – Sequential and Random access, Difference betweenboth, how to read and write data in both the cases, for random access **Pre-processor Directives**– Types of pre-processor directives, # define, #include, #undef, #line, pragma directives, Conditional directives, Predefined identifiers, Type Qualifiers, variable length arguments. recursion	20%

3. Course Outcomes (COs):

NO	Course Objective
CO1.	To recall facts, concepts, and basic syntax in C programming.
CO2.	To comprehend the meaning of code snippets, algorithms, and data structures in C.
CO3.	To use knowledge and skills to solve problems and write functional code in C.
CO4	To break down code, identify errors, and understand the flow of control in programs.
CO5.	To judge the quality, efficiency, and effectiveness of a C program using various testing techniques. To design and develop original programs, algorithms, and solutions using C.

4. **Course Outcome Mapping with Program Outcome**

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2		2				2
CO2	3	3	3	2				2
CO3	3	3	3					2
CO4	2	2	2	3				2
CO5	2	3	2				3	3

5. **Text Books:**
1) Pradip Dey – Manas Ghosh, Programming in C Second Edition Oxford Publication).
2) Reema Thareja, "Programming in C',2nd Edition, Oxford University Press.
3) Balagurusamy, Programming in ANSI C , Tata McGrew Hill.
4) Ashok N Kamthane, Programming with ANSI and Turbo C, Pearson Education.
5) Brian W. Kernigham, Dennis Ritchie, "The C Programming Language", Pearson
6) Yeshvant Kanetkar, "Let Us C", BPB Publication

6. **Webilography:**
1) 'C' Programming Language: http://www.w3schools.in/cprogramming-language/intro/
2) Learn C Online: http://www.learnconline.com/
3) 'C' Frequently Asked Questions: http://www.c-faq.com
4) 'C' Programming: http://www.cprogramming.com
5) Sams Teach Yourself C in 24 Hours: http://aelinik.free.fr/c/

7. **Accomplishment of the student after completing the course**:
After completion of the course students should become capable of solving problems using computers through C programming language.

Course Code	240110102			
Category	Core Subject			
Course Title	Relational Database Management System (RDBMS)			
Scheme and Credits	Theory	Tutorial	Lab	Credits
	4	0	0	4
Pre-requisites (if any)	Basic knowledge of data storing and retrieving mechanism in computer and types of software and applications using different types of data to be stored and retrieve as per the need. Proficiency in any programming language. The course teaching language is English, so students have to have communication, reading and apprehension skills of English.			

1. **Course Objectives:**

1	To understand the fundamental concepts of database systems, their architecture, types, and languages, and to differentiate between traditional file systems and modern DBMS/RDBMS approaches including an overview of NoSQL and NewSQL technologies.
2	To develop the ability to design and interpret Entity-Relationship (ER) models and effectively convert them into relational schemas for real-world applications, ensuring a strong foundation in logical database design.
3	To understand the principles of relational schema design, apply normalization techniques using functional dependencies, and ensure data integrity by eliminating anomalies through proper constraints and Codd's rules.
4	To understand various file organization and indexing techniques, and to analyze query execution strategies through cost estimation and optimization methods for efficient database performance.
5	To gain knowledge of transaction management concepts, including ACID properties, concurrency control, and recovery techniques, ensuring consistency and reliability in multi-user database environments.

Syllabus for two Years
School of Computer Applications,
Master of Computer Applications (MCA)
Semester - I

2. **Course Contents**

Unit	Course Content	Weight age
Unit I	**Introduction to Database System (CO1)** Data, Information, File-based Data Management, Difference between file-system and database management system, Differentiate between Database and Database Management System, Concept of Data dictionary in DBMS **Database**: Users, Data Models, Data abstraction, ANSI/SPARC Architecture, Logical and Physical data independence, DBMS Languages: DDL, DML, DCL (definitions only) **RDBMS**: Examples of RDBMS, Classifications of Database Management Systems – Relational and Non-Relational Database, Benefits of using RDBMS, Introduction to NoSQL and NewSQL (overview only)	20%
Unit II	**Entity Relationship Modelling (CO2)** **Components of an E-R Model:** Entity, Entity Sets, Attributes and keys – Super Key, Primary Key, Candidate Key, Alternate Key, Relationship Types, Composite entities, Weak entity Relational Database design by ER to Relational Mapping **Examples for ER Diagram:** College Management System, Railway Reservation System, Hospital Management System, Library Management System, e-Business Management System (can take any eBusiness example like online shopping, online food ordering etc.…)	20%

Unit III	**Relational Database Design (CO3)** Design process steps for Relational Schema, Codd's Rules, Database Relationships and Constraints – Primary Key, Foreign Key, Null, Not Null, Unique **Design Guidelines for Relational Schema:** Database design anomalies, Functional Dependencies, Normalization process – understanding Normal Forms based on Primary keys and converting database tables to 1NF, 2NF and 3NF, General Definitions with one example for Boyce-Codd Normal Forms (BCNF), Overview of 4th and 5th normal forms	**20%**
Unit IV	**Indexing, File Organization, Query Optimization (CO4)** File Organization: Heap, Sequential, Hashed , Indexing: Primary, Secondary, Composite, B+ Trees, Cost Estimation Basics, Query Optimization Techniques, Query Execution Plan, Statistics and Histograms	**20%**
Unit V	**Transaction and Concurrency Control (CO5)** Transactions and ACID Properties, Transaction States and Types, Schedules and Serializability, Concurrency Control Protocols, Deadlock Management, Recovery and Logging Techniques, Commands: Commit, Rollback, Savepoint	**20%**

3. **Course Outcomes (COs):**
   - CO1: Understand the fundamental concepts, architecture, and types of database systems, and differentiate between file-based systems and modern DBMS/RDBMS models, including NoSQL and NewSQL.
   - CO2: Apply the concepts of ER modeling to represent real-world problems and convert ER diagrams into relational schemas effectively.
   - CO3: Design efficient and anomaly-free relational schemas by applying normalization principles and enforcing constraints based on functional dependencies.
   - CO4: Analyze different file organization and indexing techniques and implement basic query optimization strategies to enhance database performance.
   - CO5: Demonstrate an understanding of transaction management, concurrency control protocols, and recovery mechanisms to maintain database consistency and integrity in multi-user environments.

4. **Course Outcome Mapping with Program Outcome:**

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2		2				2
CO2	3	3	3	2				2
CO3	3	3	3					2
CO4	2	2	2	3				2
CO5	2	3	2				3	3

5. **Text Books:**
   7) C. J. Date, A. Kannan, S. Swaminathan – "An Introduction to Database Systems", Pearson Edition, 8th Edition
   8) Ramez Elmasari and Shamkant B. Navathe – "Fundamentals of Database Systems", Pearson Edition, 7th Edition
   9) S.K. Singh – "Database Systems – Concepts, Design and Applications", Pearson Edition
   10) Abraham Silberschatz, Henry F. Korth and S. Sudarshan – "Database System Concepts", McGraw Hill International Edition, 5th Edition

6. **Web Resources:**
   1) https://docs.oracle.com
   2) https://www.tutorialspoint.com
   3) https://www.educba.com/

7. **Accomplishment of the student after completing the course:**
   After completion of the course students should become capable of designing a relational database needed for any software development. They can write optimized queries and can perform transaction management for any software system.

Syllabus for two Years
School of Computer Applications,
Master of Computer Applications (MCA)
Semester - I

Course Code	240110103			
Category	Core Subject			
Course Title	**Database Management System – Lab**			
Scheme and Credits	**Theory**	**Tutorial**	**Lab**	**Credits**
	0	**1**	**6**	**4**
Pre-requisites (if any)	Students are expected to have a basic understanding of the English language and introductory-level knowledge of computer programming. Familiarity with operating systems, file handling, and using software tools.			

1. **Course Objectives:**

1	Understand relational database concepts and basic SQL syntax.
2	Apply SQL commands for data definition, manipulation, and control
3	Perform advanced SQL operations like joins, views, sub queries, and indexes
4	Develop PL/SQL programs using procedures, functions, and triggers
5	Gain a foundational understanding of NoSQL databases and explore MongoDB for handling non-relational data.

2. **Course Contents**

Unit	Course Content	Weightage
Unit I	**Introduction to SQL** Introduction to SQL Standards, Environment & Data Types, Creating Tables, Modifying Tables, Constraints (Primary, Foreign, Not Null, Unique, Check) **Basic SQL Commands**: SELECT, INSERT, UPDATE, DELETE, WHERE, ORDER BY, DISTINCT, LIKE, IN, BETWEEN, **Aggregate Functions**: COUNT, SUM, AVG, MIN, MAX **Grouping and Filtering**: GROUP BY, HAVING	**20%** **(CO1)**

Unit II	**Advanced SQL Operations:** Set Operations: UNION, INTERSECT, MINUS Subqueries (Single-row, Multi-row, Correlated) Joins: Inner Join, Outer Joins, Self Join, Cross Join Views: Creating and Managing Views Indexes: Types and Use in Optimization Transaction Control: COMMIT, ROLLBACK, SAVEPOINT	**20%** **(CO2)**
Unit III	**Introduction to PL/SQL** Basics of PL/SQL Block Structure, Variables, Constants, Data Types, and Operators Control Structures: Conditional Statements: IF, IF-THEN-ELSE, CASE Loops: FOR, WHILE, LOOP...EXIT Cursors: Implicit and Explicit Cursors	**20%** **(CO3)**
Unit IV	**Advanced PL/SQL** Named PLSQl Block: Stored Procedures and Functions Error Handling: EXCEPTION block Triggers: Before and After, Row-level and Statement-level	**20%** **(CO4)**
Unit V	**Introduction to NoSQL** Introduction to JSON and BSON CRUD operations in MongoDB Understanding Collections and Documents Basic MongoDB shell queries	**20%** **(CO5)**

3. Course Outcomes (COs):

CO No.	Course Outcomes
CO1	Demonstrate the ability to create and manipulate relational databases using basic SQL commands.

CO No.	Course Outcomes
CO2	Apply advanced SQL features including joins, subqueries, views, indexes, and transactions.
CO3	Write PL/SQL blocks using variables, control structures, and cursors.
CO4	Develop stored procedures, functions, and triggers using advanced PL/SQL features.
CO5	Explain the fundamental concepts of NoSQL and use MongoDB to model and manipulate document-based data.

4. Course Outcome Mapping with Program Outcome:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2		2				2
CO2	3	3	2	3				2
CO3	3	2	2	2				2
CO4	3	3	3	3				2
CO5	2	2	2	3				3

5. Text Books:
11) SQL, PL/SQL: The Programming Language of Oracle by Ivan Bayross, BPB Publications
12) Oracle PL/SQL Programming by Steven Feuerstein, O'Reilly Media
13) SQL and Relational Theory: How to Write Accurate SQL Code by C.J. Date
14) Fundamentals of Database Systems by Ramez Elmasri and Shamkant Navathe, Pearson edition

6. Web Resources:
1) https://www.w3schools.com/sql – SQL tutorials

2) https://www.tutorialspoint.com/plsql – PL/SQL basics and examples
3) https://docs.oracle.com/en/database/ – Oracle official documentation
4) https://sqlzoo.net – Interactive SQL practice
5) https://www.geeksforgeeks.org/sql-tutorial/ – SQL & PL/SQL coding tutorials

7. **Accomplishment of the student after completing the course:**
After completion of the course students should become capable of designing a relational database needed for any software development. They can write optimized queries and can perform transaction management for any software system.

Syllabus for two Years
School of Computer Applications,
Master of Computer Applications (MCA)
Semester - I

Course Code	240119104			
Category	Core Subject (Elective-I)			
Course Title	Web Development Using Advanced JavaScript (WDJ)			
Scheme and Credits	Theory	Tutorial	Lab	Credits
	0	1	6	4
Pre-requisites (if any)	No			

1. Course Objectives:

1	To understand JavaScript as a dynamic scripting language and its role in web development.
2	To apply functions, object-oriented programming, and DOM manipulation in building interactive web applications
3	To explore built-in objects, browser APIs, and event-driven programming for dynamic user experiences.
4	To implement modern JavaScript features including ES6+, asynchronous programming, and data storage techniques.
5	To gain introductory experience with ReactJS and apply it through a self-paced web development project.

2. Course contents:

Module	Content	Weightage
Unit I	**JavaScript Basics:** o Evolution and features of JavaScript o Role of JavaScript in modern web development o Variables (var, let, const) and data types o Operators, control structures (if, switch, for, while, do-while) o Writing and embedding JavaScript (inline, internal, external) o Basics of integrating JavaScript with HTML	20%
Unit II	**Functions, Objects, and DOM Manipulation:** o Function declarations, expressions, arrow functions, default parameters o Recursion and closures	20%

	o Objects and object creation patterns (literals, constructors, ES6 classes) o Prototype-based inheritance and property access o Document Object Model (DOM) o Accessing and modifying elements, styles, and content o Event handling (click, mouseover, change, etc.).	
Unit III	**Built-in Objects and Browser APIs** o Built-in objects: Array, String, Math, Date, RegExp o Browser objects: Window, Document, Navigator, History o Pop-ups: alert, prompt, confirm o Timers: setTimeout, setInterval o Event-driven programming: mouse and keyboard events o Real-time interactivity and UI feedback techniques	**20%**
Unit IV	**Advanced JavaScript and Modern Features** – o Asynchronous programming: callbacks, Promises, async/await o Fetch API and AJAX basics o Error handling: try-catch-finally o ES6+ features: modules (import/export), iterators, generators o Collections: Map, Set o Web storage: localStorage, sessionStorage, cookies o Overview of modern frameworks: React, Node.js basics	**20%**
Unit V	**Introduction to ReactJS and Self-Paced Mini Project** o What is ReactJS? o JSX syntax and components o Props and state management o React hooks: useState, useEffect o Component lifecycle and conditional rendering o Creating single-page interfaces o Mini Project: Design and build a small interactive web app using ReactJS	**20%**

3. Course Outcomes (COs):

NO	Course Outcomes
CO1.	Explain the features of JavaScript and its role as a dynamic scripting language in modern web development.
CO2.	Apply functions, object-oriented programming, and DOM manipulation to build interactive web pages.

CO3.	Utilize built-in JavaScript objects, browser APIs, and event-driven programming for dynamic client-side functionality.
CO4	Implement asynchronous programming techniques, and client-side storage using modern JavaScript.
CO5.	Develop and demonstrate a small-scale front-end project using ReactJS as a component-based framework.

4. Course Outcome Mapping with Program Outcome

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1: Understand JavaScript as a dynamic scripting language and its role in web development.	3	2	2	2	2	–	–	–
CO2: Apply functions, OOP, and DOM manipulation to build interactive web applications.	3	3	3	2	2	–	–	–
CO3: Explore built-in objects, browser APIs, and event-driven programming.	3	2	2	3	2	–	–	–
CO4: Implement asynchronous programming, and client-side data storage.	3	2	2	3	3	–	–	–
CO5: Gain introductory experience with ReactJS and apply it through a self-paced web development project.	3	2	3	3	3	3	2	2

5. Text Books:
1) Eloquent JavaScript (4th Edition) – Marijn Haverbeke
2) Professional JavaScript for Web Developers (4th Edition) – Nicholas C. Zakas
3) JavaScript: The Definitive Guide (7th Edition) – David Flanagan

6. Webilography:
1) Modern JavaScript Tutorial – https://javascript.info
2) MDN Web Docs – JavaScript –
https://developer.mozilla.org/en- US/docs/Web/JavaScript
3) W3Schools JavaScript Tutorial – https://www.w3schools.com/js/
4) React Official Docs – https://reactjs.org/docs/getting-started.html
5) FreeCodeCamp JavaScript + React Courses – https://www.freecodecamp.org

7. **Accomplishment of the student after completing the course**:
   After completing the Web Development Using Advanced JavaScript course, the student will be able to develop dynamic, interactive web applications using modern JavaScript features and tools. They will also gain hands-on experience in building and deploying real-world projects, preparing them for frontend development roles.

Course Code	240119105			
Category	Elective Subject			
Course Title	Introduction to Digital Media and Content Creation(DMCC)			
Scheme and Credits	Theory	Tutorial	Lab	Credits
	0	1	6	4
Pre-requisites (if any)	Basic knowledge of computer programming. Concept of variables, data types, condition statements and loop must be clear.			

1. **Course Objectives:**

1	To understand the landscape and impact of digital media in the modern world.
2	To learn fundamental techniques for content creation tailored for digital platforms.
3	To gain hands-on experience in designing engaging visuals using graphic tools.
4	To create and edit audio/video content for publishing on digital platforms.
5	To use content publishing tools and basic analytics to manage and assess content performance.

2. **Course Contents**

Unit	Course Content	Weightage
Unit I	**Fundamentals of Digital Media (CO1)** Digital media: Definition, types, and scope, Traditional vs. digital media: Key differences Digital platforms: Social media, OTT, Mobile, Web, AR/VR Ethical and legal issues in digital content Case studies of impactful digital media campaigns	20%

Unit II	**Foundations of Content Creation (CO2)** Types of content: Text, image, video, audio, Basics of storytelling and content flow, Writing effective content for blogs and social platforms, Understanding target audience and personas, Overview of content creation tools: WordPress, Canva, Grammarly	**20%**
Unit III	**Visual Design and Image Editing (CO3)** Principles of design: Color, layout, balance, hierarchy Graphic design tools: Canva, Photoshop, Illustrator Creating social media creatives, posters, and banners Brand consistency and visual identity Image resolution, optimization, and file formats	**20%**
Unit IV	**Audio & Video Content Development (CO4)** Scriptwriting, storyboarding, and planning, Basics of video shooting and editing Tools: CapCut, Adobe Premiere, Filmora, Audio recording/editing with Audacity Creating YouTube Shorts, Instagram Reels, podcasts	**20%**
Unit V	**Publishing, Promotion, and Analytics (CO5)** Content publishing platforms: YouTube, Instagram, WordPress SEO for content creators Social media scheduling tools: Hootsuite, Buffer Introduction to Google Analytics, Meta Insights Case studies of viral campaigns and content strategies	**20%**

3. Lab/Practical Component

Week	Practical Exercise	Tools Used	Related CO
1	Write a blog post for a target audience	Google Docs / WordPress	CO2
2	Design a poster and social media creative	Canva / Photoshop	CO3

Week	Practical Exercise	Tools Used	Related CO
3	Record and edit a voiceover or podcast intro	Audacity	CO4
4	Shoot and edit a short video (30-60 sec)	CapCut / Filmora	CO4
5	Publish a post on a dummy blog and analyze performance	WordPress / Google Analytics	CO5

4. Course Outcomes (COs):

CO Code	Course Outcome Statement
CO1	Understand the fundamentals, types, scope, and ethical considerations of digital media and analyze the impact of digital platforms.
CO2	Apply foundational techniques of storytelling and content creation tailored to diverse platforms and audiences using suitable tools.
CO3	Design visually engaging digital content by applying core design principles using modern graphic tools.
CO4	Develop and edit audio and video content for various digital platforms using appropriate editing tools and techniques.
CO5	Use publishing tools, promote content using SEO and scheduling tools, and analyze content performance using analytics platforms.

5. Course Outcome Mapping with Program Outcome:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2					2	2
CO2	3	2	2	2				2
CO3			3	3				2
CO4		2	3	3	2			2
CO5	2		3	3	2	2	2	2

6. Text Books:

1) Digital Media and Society – Adrian Athique, Polity Press

2) Content Rules – Ann Handley & C.C. Chapman, Wiley
3) The Art of Social Media – Guy Kawasaki & Peg Fitzpatrick
4) YouTube Secrets – Sean Cannell & Benji Travis
5) Adobe Photoshop Classroom in a Book – Adobe Press

7. Web Resources:
4) Google Digital Garage – Free digital marketing course
5) HubSpot Academy – Free content marketing and SEO training
6) Canva Design School
7) Meta Blueprint

8. Accomplishment of the student after completing the course:
Upon completing this course, students will be proficient in designing and managing relational databases using SQL and PL/SQL. They will also gain foundational skills in working with NoSQL databases like MongoDB, enabling them to handle modern, document-oriented data requirements.

Course Code	240110106			
Category	Core Subject			
Course Title	Skill Enhancement Using AI Tools (SET)			
Scheme and Credits	Theory	Tutorial	Lab	Credits
	0	1	6	4
Pre-requisites (if any)	Basic computer literacy, communication skills, and familiarity with online tools are required as prerequisites.			

1. **Course Objectives:**

1	To understand the relevance and scope of AI tools in enhancing academic and professional productivity.
2	To learn to use AI-based content creation and design tools effectively.
3	To develop skills to automate tasks, generate insights, and solve problems using AI platforms.
4	To gain hands-on experience with AI tools for code generation, summarization, and presentations.
5	To apply AI tools to build personal productivity workflows and prepare industry-ready portfolios.

2. Course Contents

Unit	Course Content	Weightage
Unit I	**Introduction to AI Tools and Academic Applications** o Overview of Artificial Intelligence and its real-life use cases in academia and industry. o Benefits of using AI tools for enhancing productivity, learning, and automation. o Tool 1: NotepadAI – AI-powered writing assistant designed to help users quickly generate, summarize, and organize content with ease. o Tool 2: Grammarly – Real-time grammar, spell check, tone suggestion, and clarity improvements for academic and professional writing. o Hands-on sessions on creating summaries and improving academic reports using these tools. Ideal for notes, blogs, and professional writing tasks	**20%**
Unit II	**AI for Visual and Content Design** o Understanding AI-assisted creativity and the role of visuals in communication. o Tool 3: Canva (with Magic Write and AI templates) – Creating visually engaging posters, resumes, infographics, and academic presentations. o Tool 4: Beautiful.ai – Smart presentation creation with automated formatting and visual storytelling. o Use cases in classroom presentations, digital posters, and content branding.	**20%**
Unit III	**AI for Task Automation and Content Generation** o Leveraging AI for personal productivity and workflow management. o Tool 5: Notion AI – Organizing tasks, generating content, and summarizing notes in collaborative digital workspaces. o Tool 6: Copy.ai – AI-powered platform for generating blogs, project descriptions, and assignment drafts. o Applications in academic documentation, blog creation, and team project coordination.	**20%**
Unit IV	**Coding and Technical Concept Simplification** o Introduction to AI in software development and	**20%**

	educational support. ○ Tool 7: GitHub Copilot – Real-time code suggestions and explanations directly inside IDEs (e.g., VS Code). ○ Tool 8: ExplainLikeImFive (ELI5.ai) – Breaking down complex coding and theoretical concepts into plain language. ○ Use cases: writing logic, fixing code, explaining algorithms in classroom-friendly formats.	
Unit V	**Career Readiness and Portfolio Creation using AI** ○ Exploring AI for self-branding and career development. ○ Tool 9: Resume.io / Kickresume AI – Auto-generate personalized resumes, cover letters, and portfolios with smart formatting. ○ Tool 10: Pictory.ai – Convert blog posts, scripts, or reports into engaging AI-generated videos for digital portfolios or presentations. ○ Mini Project: Students will select 2–3 tools from the syllabus to create a personal or academic portfolio that demonstrates their AI tool proficiency.	**20%**

3. Lab/Practical Component.

Week	Practical Exercise	AI Tool Used	Related CO
1	Generate a study plan or research outline for a topic	NotepadAI	CO1
2	Proofread and improve the tone of an assignment/report	Grammarly	CO1
3	Design a poster or banner for an academic event	Canva	CO2
4	Create an AI-powered academic presentation	Beautiful.ai	CO2
5	Organize project tasks and summarize class notes	Notion AI	CO3

6	Generate introductory content for a blog or project	Copy.ai	CO3
7	Use AI to write and test a simple code snippet	GitHub Copilot	CO4
8	Explain a programming concept in simple terms	ExplainLikeImFive	CO4
9	Create an AI-generated resume with portfolio links	Resume.io / Kickresume AI	CO5
10	Convert a blog post or project script into a video	Pictory.ai	CO5

4. Course Outcomes (COs):

CO Code	Course Outcome Statement
CO1	Understand the relevance of AI tools and apply them to enhance academic productivity through content writing and improvement tools like NotepadAI and Grammarly.
CO2	Use AI-powered tools such as Canva and Beautiful.ai to create visually engaging and impactful academic and professional content.
CO3	Automate routine tasks and generate creative content using productivity platforms like Notion AI and Copy.ai.
CO4	Apply AI-based coding assistants and explainers to simplify programming and technical learning using tools like GitHub Copilot and ELI5.ai.
CO5	Utilize AI tools to build resumes, portfolios, and create digital assets for career advancement and industry readiness.

5. Course Outcome Mapping with Program Outcome:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	2	2		3			2	3
CO2	2		2	3				2
CO3	2	2	3	3	2			3
CO4	3	2	3	3				2
CO5	2		3	3	2	2	2	3

6. **Web Resources:**
   • NotepadAI – https:// notepad-plus-plus.org , https://www.chatgptnotepad.com
   • Grammarly – https://www.grammarly.com
   • Canva – https://www.canva.com
   • Beautiful.ai – https://www.beautiful.ai
   • Notion AI – https://www.notion.so/product/ai
   • Copy.ai – https://www.copy.ai
   • GitHub Copilot – https://github.com/features/copilot
   • ExplainLikeImFive – https://eli5.gg
   • Resume.io – https://resume.io / Kickresume – https://www.kickresume.com
   • Pictory.ai – https://pictory.ai

7. **Accomplishment of the student after completing the course:**
   After completing the course, students will be able to effectively use AI tools to enhance productivity, automate tasks, and create industry-ready content and portfolios.