SYLLABUS

for

Bachelor of Science

(2025 Onwards)

(Computer Science as a One Subject)

Programme

As Per

New Education Policy (NEP-2020)

HEMVATI NANDAN BAHUGUNA GARHWAL UNIVERSITY

A CENTRAL UNIVERSITY

SRINGAR GARHWAL – (246174)

PROGRAM STRUCTURE

SEMESTER – I

TYPE	PAPER CODE		PAPER NAME
MAJOR	(SOS/CS/MJ/C101)	:	Programming In C
MINOR	(SOS/CS/MI/C102)	:	Programming In C
MD/ID	(SOS/CS/MD/C103)	:	Fundamental of Information Technology
SKILL		:	ELECTIVE – I
MAJOR Lab	(SOS/CS/MJ/CP11)	:	Programming In C – Lab
MINOR Lab	(SOS/CS/MI/CP12)	:	Programming In C – Lab
MD/ID Lab	(SOS/CS/MD/CP13)	:	Fundamental of Information Technology – Lab

SEMESTER – II

TYPE	PAPER CODE		PAPER NAME
MAJOR	(SOS/CS/MJ/C201)	:	Data Structure Using C
MINOR	(SOS/CS/MI/C202)	:	Data Structure Using C
MD/ID	(SOS/CS/MD/C203)	:	Internet of Things
SKILL		:	ELECTIVE – I
MAJOR Lab	(SOS/CS/MJ/CP21)	:	Data Structure Using C – Lab
MINOR Lab	(SOS/CS/MI/CP22)	:	Data Structure Using C – Lab
MD/ID Lab	(SOS/CS/MD/CP23)	:	Internet of Things – Lab

SEMESTER – III

TYPE	PAPER CODE		PAPER NAME
MAJOR	(SOS/CS/MJ/C301)	:	Object Oriented Programming Using C++
MINOR	(SOS/CS/MI/C302)	:	Numerical Computing - I
MD/ID	(SOS/CS/MD/C303)	:	Introduction to Cyber Security
SKILL	(SOS/CS/SEC304)	:	Digital Electronics - I
MAJOR Lab	(SOS/CS/MJ/CP31)	:	Object Oriented Programming Using C++ Lab
MINOR Lab	(SOS/CS/MI/CP32)	:	Numerical Computing - I Lab
MD/ID Lab	(SOS/CS/MD/CP33)	:	Cyber Security Case Studies

PROGRAM STRUCTURE

SEMESTER – IV

TYPE	PAPER CODE		PAPER NAME
MAJOR	(SOS/CS/MJ/C401)	:	Design and Analysis of Algorithms
MINOR	(SOS/CS/MI/C402)	:	R Programming
MD/ID	(SOS/CS/MD/C403)	:	E – Commerce
SKILL	(SOS/CS/SEC404)	:	Mathematical Foundation of Computer Science
MAJOR Lab	(SOS/CS/MJ/CP41)	:	Design and Analysis of Algorithms – Lab
MINOR Lab	(SOS/CS/MI/CP42)	:	R Programming Lab
MD/ID Lab	(SOS/CS/MD/CP43)	:	E- Commerce Case Studies

SEMESTER – V

TYPE	PAPER CODE		PAPER NAME
MAJOR – 1	(SOS/CS/MJ/C501)	:	Operating System
MAJOR – 2	(SOS/CS/MJ/C502)	:	Computer Graphics
MAJOR ELECTIVE		:	ELECTIVE – II
MINOR **SKILL/ VOCATIONAL**		:	ELECTIVE – III
MAJOR 1 Lab	(SOS/CS/MJ/CP51)	:	Basic Unix – Lab
MAJOR 2 Lab	(SOS/CS/MJ/CP52)	:	Computer Graphics – Lab
VOCATIONAL		:	ELECTIVE IV

SEMESTER – VI

TYPE	PAPER CODE		PAPER NAME
MAJOR – 1	(SOS/CS/MJ/C601)	:	Basic Java Programming
MAJOR – 2	(SOS/CS/MJ/C602)	:	Database Management System
MAJOR ELECTIVE		:	ELECTIVE – V
MINOR **SKILL/ VOCATIONAL**		:	ELECTIVE – VI
MAJOR 1 Lab	(SOS/CS/MJ/CP61)	:	Basic Java Programming – Lab
MAJOR 2 Lab	(SOS/CS/MJ/CP62)	:	Database Management Systems – Lab
PROJECT		:	----

PROGRAM STRUCTURE

B.Sc. 4th Year (U.G. with Honours)

SEMESTER – VII

TYPE	PAPER CODE		PAPER NAME
Major 1	(SOS/CSE/MJ/C701)	:	Computer Organization and Architecture
Major 2	(SOS/CSE/MJ/C702)	:	Theory of Computation
Major 3	(SOS/CSE/MJ/C703)	:	Advance Operating Systems
Major Elective		:	Elective VII
Minor	(SOS/CSE/MI/C704)	:	Natural Language Processing
Major Practical	(SOS/CSE/MJ/CP705)	:	Shell Programming

SEMESTER – VIII

TYPE	PAPER CODE		PAPER NAME
Major 1	(SOS/CSE/MJ/C801)	:	Compiler Design
Major 2	(SOS/CSE/MJ/C802)	:	Network Security and Cryptography
Major 3	(SOS/CSE/MJ/C803)	:	Software Engineering
Major Elective		:	Elective VIII
Minor	(SOS/CSE/MI/C804)	:	Data Mining
Major Practical	(SOS/CSE/MJ/CP805)	:	Compiler Design Lab

B.Sc. 4th Year (U.G. Honours with Research)

SEMESTER – VII

TYPE	PAPER CODE		PAPER NAME
Major 1	(SOS/CSE/MJ/C701)	:	Computer Organization and Architecture
Major 2	(SOS/CSE/MJ/C702)	:	Theory of Computation
Major*	(SOS/CSE/MJ/R703)	:	Research Methodology
Major Elective		:	Elective VII
Minor	(SOS/CSE/MI/C704)	:	Natural Language Processing
Major Practical	(SOS/CSE/MJ/CP705)	:	Digital Electronics Lab

PROGRAM STRUCTURE

SEMESTER – VIII

TYPE	PAPER CODE		PAPER NAME
Major 1	(SOS/CSE/MI/C704)	:	Compiler Design
Major Elective		:	Elective VIII
Major*		:	Dissertation
Minor	(SOS/CSE/MI/C804)	:	Data Mining
Major Practical	(SOS/CSE/MJ/CP805)	:	Compiler Design Lab

SEC/Vocational/Elective Course Bucket

ELECTIVE – I	ELECTIVE – II	ELECTIVE – III
A. Office Automation (SOS/CS/SEC/E101)	**A.** Computer Networks (SOS/CS/MJ/E201)	**A.** Probability and Statistics (SOS/CS/MI/E301)
B. Web Development (SOS/CS/SEC/E102)	**B.** Blockchain & It's Application (SOS/CS/MJ/E202)	**B.** Cyber Security and Privacy (SOS/CS/MI/E302)
	C. MOOC (SWAYAM/NPTEL)	**C.** MOOC (SWAYAM/NPTEL)

ELECTIVE – IV	ELECTIVE – V	ELECTIVE VI
A. Programming Using C# & .NET (SOS/CS/SEC/E401)	**A.** Discrete Mathematics (SOS/CS/MJ/E501)	**A.** Artificial Intelligence (SOS/CS/MI/E601)
B. Wireless & Mobile Computing (SOS/CS/SEC/E402)	**B.** Introduction to Data Science (SOS/CS/MJ/E502)	**B.** Distributed System (SOS/CS/MI/E602)
C. MOOC (SWAYAM/NPTEL)	**C.** MOOC (SWAYAM/NPTEL)	**C.** MOOC (SWAYAM/NPTEL)

ELECTIVE – VII	ELECTIVE – VIII
A. Machine Learning (SOS/CS/MJ/E701)	**A.** Python Programming (SOS/CS/MJ/E801)
B. Cloud Computing (SOS/CS/MJ/E702)	**B.** Image Processing (SOS/CS/MJ/E802)
C. MOOC (SWAYAM/NPTEL)	**C.** MOOC (SWAYAM/NPTEL)

Course Structure and Credit Allocation for Program

The following course structure under FYUP for Multidisciplinary programmes is designed for subjects which have practical based courses or have relatively larger emphasis on practical course-based learning:

First Year (NHEQF Level-4.5)

Course Category	Semester-I					Semester-II			
	Subject/Title	No. of paper	Credits			Subject/Title	No. of paper	Credits	
			T	P				T	P
Discipline Specific Core	Programming In C **(Major)**	1	2	2		Data Structure Using C **(Major)**	1	2	2
	Programming In C **(Minor)**	1	2	2		Data Structure Using C **(Minor)**	1	2	2
MD/ID Subject-1	Fundamental of Information Technology	1	2	2		Internet of Things	1	2	2
MD/ID Subject-2	MD/ID-I	1	2	2		MD/ID-II	1	2	2
SEC/ AEC	**Elective - I** Field work/SEC/ Communication Skills **Or** AMSC/Field Work/SEC	1	2	-		**Elective - I** AMSC/Field Work/SEC **Or** Field work/ SEC/ Communication Skills	1	2	-
VAC	Understanding and Connecting with Environment **Or** Life Skills & Personality development	1	2	-		Understanding and Connecting with Environment Or Life Skills & Personality Development	1	2	-
Total		**6**	**12**	**8**			**6**	**12**	**8**
NHEQF Level-4.5	**Student on exit after successfully completing first year (i.e., securing minimum required 40 credits + 4 Credits in one Vocational Course/Skill-Enhancement Course of 4 credits) will be awarded "Undergraduate Certificate" of one year, in related field/discipline/subject.**								

- ❖ The student may opt for any one course from Field Work/ Skill Enhancement Course (SEC)/ Communication Skills in one semester, and any one course from Additional Multidisciplinary Skill Course (AMSC)/ Field Work/ Skill Enhancement Course (SEC) in the other semester.
- ❖ Field Work/Discipline Specific Skill Enhancement Course (SEC): Studeny may opt SEC/Field Work related to any discipline subject opted by her/him as a DSC in the first year.
- ❖ **Field Work:** In addition to providing students with practical, experience-based learning, field work aims to expose them to real-world socio-economic and societal challenges, allowing them to bridge the gap between theory and practice and develop effective solutions to real-life problems.
- ❖ **AMSC:** Additional Multidisciplinary Skill Course (is offered as SEC)

Following courses are offered under AMSC, University may add new courses under AMSC in future:

1. Plant Nursery Development and Management
2. Basic Yoga Practices
3. Physical Education and Sports Management
4. Regional Folklores and their Cultural Context
5. Indian Traditional Music
6. Tour and Travel Operations

- ❖ **Communication Skills (AEC):** 'Communication Skills' course will be offered in Hindi, English and Sanskrit Languages, student may opt any one language for studying the course
- ❖ **Life Skill & Personality Development (VAC)**
- ❖ **Understanding and Connecting with Environment (VAC)**

Second Year (NHEQF Level- 5)

The student will pursue one Major and One Minor from second year out of the two DSC subjects studied in the first year (i.e. I & II Semesters).

Course Category	Semester-III					Semester-IV			
	Subject/Title	No. of paper	Credits			Subject /Title	No. of paper	Credits	
			T	P				T	P
Major-I (One Subject)	Object Oriented Programming Using C++	1	4	2		Design and Analysis of Algorithms	1	4	2
Minor-I (One Subject)	Numerical Computing - I	1	2	2		Numerical Computing - II	1	2	2
SEC	Digital Electronics - I	1	2	-		Digital Electronics - II	1	2	-
MD/ID	Introduction to Cyber Security	1	2	2		E - Commerce	1	2	2
AEC (Language based courses)	Indian, Modern, Regional Language-I	1	2	-		Indian, Modern, Regional Language-II	1	2	-
VAC/AEC	IKS Or Culture, Traditions and Moral Values	1	2	-		IKS Or Culture, Traditions and Moral Values	1	2	-
Total		**6**	**14**	**6**			**6**	**14**	**6**
NHEQF Level-5	**Student on exit after successfully completing Second year (i.e., securing minimum required 80 credits + 4 Credits in one vocational course/skills-enhancement course of 4 credits) will be awarded "Undergraduate Diploma" of two year, in related field/discipline/subject.**								

- IKS-Indian Knowledge System (AEC)
- Culture, Traditions and Moral Values (VAC)
- Students are required to study both courses—Indian Knowledge System (IKS) and Culture, Traditions and Moral Values during the III and IV semesters. However, they will have the flexibility to study one course in each semester.
- Indian, Modern, Regional Language—Hindi, Sanskrit and English (Student have to study 2 different languages in the second year with one language in one semester and other language in another semester).
- The department may offer a 2-credit SEC Major course as either a fully theory-based or fully practical-based module.
- The student will pursue one Major and One Minor from second year out of the two DSC subjects studied in the first year (i.e. I & II Semesters).

Third Year (NHEQF Level-4.5)

A student pursuing a Bachelor's degree will be required to continue with the same Major and Minor subjects in the third year as selected in the second year of the four-year undergraduate programme.

Course Type	Semester-V					Semester-VI			
	Subject/Title	No. of paper	Credits			Subject /Title	No. of paper	Credits	
			T	P				T	P
Major-I (One)	Operating System	1	4	2		Basic Java Programming	1	4	2
	Computer Graphics	1	4	2		Database Management System	1	4	2
	Elective-II **(Major Elective)**	1	4	-		Elective-II **(Major Elective)**	1	4	-
	Field visit/ Vocational/ Internship	1		4		Project Report	1	-	4
Minor (One)	Elective - III **Minor-I (SEC/ Or Vocational/ Or Community Outreach)**	1	4	-		Elective - III **Minor-II (SEC/ Or Vocational/ Or Community Outreach)**	1	4	-
Total		**5**	**16**	**8**			**5**	**16**	**8**
NHEQF Level-5.5	**Student on exit after successfully completing three years (i.e., securing minimum required 128 credits will be awarded "Bachelor's Degree" of three year, in related field/discipline/subject.**								

❖ **Community Outreach-**The curricular component of 'community outreach' will involve activities that would expose students to the socio-economic issues in society so that the theoretical learnings can be supplemented by actual life experiences to generate solutions to real-life problems.

Note:

1. In case of DSE Major Elective course, if the department want to introduce practical component, the department may bifurcate the total 4 credits between theory and practical.
2. In case of Vocational course being offered by the department, the 4 credits may be given entirely to the theory course or distributed between theory and practical as per the requirements.
3. Student will continue with the same Minor in the third year (V & VI Semester) as studied in the second year (III & IV semester) of the FYUP.

Fourth Year – NHEQF Level- 6

In the fourth Year of Four-Year Undergraduate Program (FYUP) two types of programs are offered:

1. U.G. with Honours

2. U.G. Honours with Research

Candidates with a minimum CGPA of 7.5 will be eligible to continue their studies into the fourth year of the undergraduate program, leading to a four-year Bachelor's degree (Honours with Research).

Fourth Year- (U.G. with Honours)

Entry requirement	(After completing requirements of a 3-year bachelor's degree (120 credits) and 2 additional credits under SSD, will be allowed to continue studies in the fourth year of the undergraduate programme leading to the four years bachelor's degree (with Honours).							
Course Type	**Semester-VII**				**Semester-VIII**			
	Subject/Title	**No. of paper**	**Credits**		**Subject /Title**	**No. of paper**	**Credits**	
			T	**P**			**T**	**P**
Major Subject (One)	Computer Organization and Architecture **(Core Major -I)**	**1**	**5**	**-**	Compiler Design **(Core Major -I)**	**1**	**5**	**-**
	Theory of Computation **(Core Major –II)**	**1**	**5**	**-**	Network Security and Cryptography **(Core Major –II)**	**1**	**5**	**-**
	Advance Operating Systems **(Core Major –III)**	**1**	**5**		Software Engineering **(Core Major –III)**	**1**	**5**	**-**
	Elective - V **(Core Major Elective –V*)**	**1**	**4**	**-**	Elective - VI **(Core Major Elective –VI*)**	**1**	**4**	**-**
	Shell Programming **(Major Practical)**	**1**	**-**	**5**	Compiler Design Lab **(Major Practical)**	**1**	**-**	**5**
Minor (One)	Natural Language Processing **(Minor–I #)**	**1**	**2**	**2**	Data Minning **(Minor–II #)**	**1**	**2**	**2**
Total		**6**	**21**	**7**		**6**	**21**	**7**
NHEQF Level- 6	***Student on exit after successfully completing four years (i.e., securing minimum required 176 credits along with securing additional 2 credits under SSD course work) will be awarded "Four years Bachelor's Degree (Honours)", in related field/discipline***							

***Note:** * In case of Core Major Elective course, if the department want to introduce practical component, the department may bifurcate the total 4 credits between theory and practical.*

***Note:** # If the minor course is offered without a practical component, the department must allocate 4 credits to the theory component. Electives may be offered by the departments under the Minor.*

Minor–I* Each department will have to prepare Minor course (One in each semester), which enriches the learner's knowledge beyond the Major discipline (Core Major). The minor courses opted by any learner should be different from the Core Major offered by the Department.

Fourth Year- (U.G. Honours with Research)

Entry requirement	(After completing requirements of a 3-year bachelor's degree (120 credits) and 2 additional credits under SSD, candidates who meet a minimum CGPA of 7.5 will be allowed to continue studies in the fourth year of the undergraduate programme leading to the four years bachelor's degree (Honours with Research).							
Course Type	**Semester-VII**				**Semester-VIII**			
	Subject/Title	**No. of paper**	**Credits**		**Subject /Title**	**No. of paper**	**Credits**	
			T	**P**			**T**	**P**
Core Subject (One)	Computer Organization and Architecture **(Core Major -I)**	**1**	**5**	**-**	Compiler Design **(Core Major-I)**	**1**	**5**	**-**
	Theory of Computation **(Core Major –II)**	**1**	**5**	**-**				
	Elective – V **(Core Major Elective-I*)**	**1**	**4**	**-**	Elective – VI **(Core Major Elective -I*)**	**1**	**4**	**-**
	Digital Electronics Lab **(Major Practical)**	**1**	**-**	**5**	Compiler Design Lab **(Major Practical)**	**1**		**3**
	Research Methodology	**1**	**5**		Dissertation	**1**		**12**
Minor (One)	Natural Language Processing **(Minor–I #)**	**1**	**2**	**2**	Data Minning **(Minor–II #)**	**1**	**2**	**2**
Total		**6**	**21**	**7**		**5**	**11**	**17**
NHEQF Level- 6	***Student on exit after successfully completing four years (i.e., securing minimum required 176 credits along with securing additional 2 credits under SSD course work) will be awarded "Four years Bachelor's Degree (Honours with Research)", in related field/discipline***							

***Note: *** *In case of Core Major Elective course, if the department want to introduce practical component, the department may bifurcate the total 4 credits between theory and practical.*

Note: # *If the minor course is offered without a practical component, the department must allocate 4 credits to the theory component. Electives may be offered by the departments under the Minor courses.*

Minor–I* Each department will have to prepare Minor course (One in each semester), which enriches the learner's knowledge beyond the Major discipline (Core Major). The minor courses opted by any learner should be different from the Core Major offered by the Department.

If a student selects a minor course from a particular subject or department, they are required to study the courses offered by that same subject/department in both the 7th and 8th semesters.

Important Note: *The student may select Minor course either from his/her second core, studied up to 6th semester, or may select from the I.D/M.D subject they have pursued in the first and second year of their U.G. Programme.*

For Example: If a student has passed U.G. 3 years with two core subjects i.e. Computer Science and Mathematics, and the student have opted for Computer Science as his/her Major subject in the 4th year, then the student may opt Minor courses (To be studied in 7th and 8th semester) from any one subject, which could be either Mathematics or ID/MD subject studied by him/her in first two years of FYUP.

Semester I

Course: Programming in C (SOS/CS/MJ/C101) (SOS/CS/MI/C102)

Course Objective:

The Programming in C course aims to provide students with a solid foundation in programming using the C language. Participants will start by learning the basics of C syntax, data types, and control structures, progressing to more advanced topics like functions, pointers, and memory management. The course focuses on developing problem-solving skills through hands-on coding exercises and projects, enhancing participants' ability to translate real-world problems into C programs.

Course Outcome:

By the end of this course, students should be able to:

1. Understand the basic programming concepts and syntax of the C language.

2. Design and implement algorithms to solve simple programming problems.

3. Write, compile, and execute C programs using integrated development environments (IDEs).

Course Content/Syllabus:

Unit-1: Introduction to C Programming Basics of C programming language, Structure of C program, Data types, Variables, Constants, Input and Output functions, Operators,
Expressions, Control structures: if, if-else, switch.

Unit-2: Loops and Functions Looping constructs: while, do-while, for, Nested loops, Arrays, Strings, Pointers, Functions: declaration, definition, call, return, recursion.

Unit-3: Storage Classes and Preprocessor Storage Classes: auto, extern, static, register, Scope and lifetime of variables, Introduction to the Preprocessor, Macros, Conditional compilation.

Unit-4: User-defined Data Types Structures and Unions, Introduction to user-defined data types, Structures: declaration, initialization, accessing members, Arrays of structures, Pointers to structures, Unions: declaration, accessing members.

Unit-5: File Handling and Command-line Arguments File operations: Opening, Reading, Writing, and Closing files, Random Access to files, Command- line arguments, File handling functions, Error handling.

Recommended Books:

1. "C Programming Absolute Beginner's Guide" by Greg Perry and Dean Miller.
2. "The C Programming Language" by Brian W. Kernighan

Semester I

Course: Fundamentals of Information technology (SOS/CS/MD/C103)

Course Objective:

1. Take on significant positions in various IT work
2. Collaborate in diverse team environments
3. Contributions in the field of IT
4. Work effectively in the IT field to make a positive contribution to society

Course Outcome:

1. Develop information technology solutions by evaluating user requirements in the systems development environment.
2. Apply knowledge of IT requirements for technology solutions in cutting edges applications.
3. Analyze a problem and identify and define the computing requirements for the appropriate solutions.
4. Create, select and apply appropriate techniques, resources, and modern engineering and IT tools.

Course Content:

Unit 1. Introduction: Definition of Electronic Computer, Generations, Classification of Computers, Computer Hardware and Basic Computer Organization: CPU- ALU, CU; RAM/ROM, Various I/O devices, Peripherals, Storage Media.

Unit 2. Computer Languages: Binary, Hexadecimal Number System; Basic Binary Logic Operations; Binary Addition and Subtraction; Generation of Languages, Assembly Language, High level language; Translators, Interpreters, Compilers, Compilers; Flow Charts, Dataflow Diagram,

Unit 3. OS & Office: Software- System and Application Software; Elementary Concepts in Operating System; Textual Vs GUI, Introduction to DOS, MS Windows, UNIX/Linux

Unit 4. Computer Networks: Elements of Communication system; Brief Introduction to Computer Networks- Introduction of LAN and WAN. Network Topologies, Client-server Architecture, IoT, Cloud Computing

Unit 5. Internet: Internet & World Wide Web, Hypertext Markup Language, DHTML, Python, WWW, Gopher, FTP, Telnet, Web Browsers, Net Surfing, Search Engines, Email; Introduction to Web Development, Static and Dynamic Pages

Unit 6. IT Application and Multi-media: Basic Awareness of NICNET and ERNET; E Commerce, E governance; Brief Introduction to Different Formats of Image, Audio, Video Unit 7. Information Concepts & Processing Definitions of Information, Need of information, quality of information, value of information, concept of information, Entropy category and Level of information in Business Organization, Data Concepts and Data Processing, Data Science, Data Representation, Application of IT to E-commerce, Electronic Governance, Multimedia, Entertainment, Introduction to Information System.

Recommended Books:

1. Sinha, Sinha, “Computer Fundamentals”,
2. Yadav R. P. , “Information Technology”

Semester I

Elective I

Course: Office Automation (SOS/CS/SEC/E101)

Course Objective:

The "Office Automation" course aims to familiarize participants with tools and techniques for streamlining office tasks using digital technologies. By the course's end, students will be proficient in using office software applications for document creation, data analysis, and communication. The objective is to enhance productivity, collaboration, and efficiency in modern office environments through effective utilization of automation tools.

Course Outcome:

By the end of this course, students should be able to:

1. Effectively use office automation tools for document creation, presentation, and data analysis.
2. Apply advanced features of office software to optimize office tasks and workflows.
3. Demonstrate effective communication and collaboration skills using office automation tools.

Course Content/Syllabus:

Unit-1: Introduction to Office Automation Overview of office automation and its benefits, Office Suite applications and their features.

Unit-2: Word Processing and Spreadsheets Word processing: Formatting, Styles, Mail Merge, Spreadsheets: Formulas, Functions, Charts.

Unit-3: Presentation Tools and Email Management Creating and delivering presentations, Email and Calendaring: Managing emails and appointments.

Unit-4: Database Management and Workflow Automation Database management: Creating and querying databases, Document collaboration and version control, Workflow automation using macros and scripts.

Recommended Books:

1. "Microsoft Office 365 & Office 2019: Introductory" by Misty E. Vermaat.
2. "Office 2019 All-in-One for Dummies" by Peter Weverka.

Semester I

Course: Web Development (SOS/CS/SEC/E102)

Course Objectives

1. Understand the fundamentals of HTML for creating structured web pages.

2. Learn CSS for styling and layout of web pages.

3. Build static, responsive, and visually appealing websites.

4. Gain a foundation for further learning in JavaScript and advanced web technologies.

Course Outcomes

1. Create structured web pages using HTML tags.

2. Apply CSS properties for styling, colors, and layouts.

3. Design responsive web pages using CSS positioning and media queries.

4. Develop a static website combining HTML and CSS effectively.

Unit	Topics
Unit 1: Introduction to Web Development	Basics of the Internet, WWW, Browser, Web Servers, Overview of HTML, CSS, and their roles in web development.
Unit 2: HTML Basics	Structure of HTML document, Basic Tags (<html>, <head>, <title>, <body>), Text formatting (<h1>–<h6>, <p>, <b>, <i>, <u>), Lists, Links, Images.
Unit 3: Tables and Forms in HTML	Creating tables (<table>, <tr>, <td>, <th>), Table attributes, Merging cells, Creating forms (<form>, <input>, <textarea>, <select>, <button>).
Unit 4: CSS Fundamentals	Introduction to CSS, Inline, Internal, and External CSS, Syntax and Selectors (element, class, id), Colors, Fonts, Text Styling, Box Model (margin, padding, border).
Unit 5: CSS Layouts and Positioning	Background properties, Display property, Positioning (static, relative, absolute, fixed), Flexbox, Grid Layout, Navigation bars, Z-index.
Unit 6: Responsive Web Design	Media Queries, Viewport meta tag, Relative units (em, %, vh, vw), CSS units for responsiveness.
Unit 7: Advanced CSS Features	CSS Transitions, Animations, Pseudo-classes (:hover, :nth-child), Pseudo-elements (::before, ::after), CSS Variables.

Recommended Books:

1. HTML & CSS: Design and Build Websites – Jon Duckett
2. Learning Web Design – Jennifer Niederst Robbins

Semester I

Lab Manual

Programming in C Lab: SOS/CS/MJ/CP11 and SOS/CS/MI/CP12:

Course Content:

1. Implementation of basic C programs using formatted I/O operations, data types, and evaluation of arithmetic expressions.
2. Implementation of decision-making and branching statements using if-else, nested if, and switch-case structures.
3. Implementation of loop control structures using while, do-while, and for loops for iterative problem-solving.
4. Implementation of 1D and 2D arrays for operations like searching (Linear and Binary) and matrix manipulation (Addition and Multiplication).
5. Implementation of modular programming using user-defined functions, passing arguments by value and by reference.
6. Implementation of recursive functions, storage classes (auto, extern, static, register), and string handling operations.
7. Implementation of advanced memory concepts using pointers, pointer arithmetic, and dynamic memory allocation (malloc, calloc, realloc, free).
8. Implementation of user-defined data types using structures and unions, and file handling operations (reading and writing text files).

SOS/CS/MD/CP13: Fundamental of Information Technology Lab

Course Content:

1. Implementation of basic operating system commands in Windows/Linux environment for file and directory management.
2. Implementation of advanced document formatting, mail merge, table generation, and macro creation using Word Processing software.
3. Implementation of data analysis worksheets using Spreadsheet software, incorporating complex formulas, built-in functions, and pivot tables.
4. Implementation of data visualization techniques by generating column, bar, line, and pie charts within spreadsheet software.
5. Implementation of multimedia presentations using Presentation software, applying slide transitions, animations, and hyperlinks.
6. Implementation of basic database management tasks using MS Access or similar tools, including table creation, primary keys, and relationships.

Semester II

Course: Data Structure (SOS/CS/MJ/C201) (SOS/CS/MI/C201)

Course Objective:

The objective of this course is to familiarize students with fundamental data structures and their applications in solving computational problems. The course aims to develop students' skills in designing, implementing, and analysing various data structures

Course Outcome:

By the end of this course, students should be able to:

1. Understand the concepts of data structures and their importance in computing.
2. Implement and use common data structures such as arrays, linked lists, stacks, queues, trees, and graphs.
3. Analyze the time and space complexity of algorithms related to data structures.
4. Apply data structures to efficiently solve real-world problems.

Course Content:

Unit-1: Introduction to Data Structures Overview of data organization and data types, Arrays, Linked Lists, and Stacks, Queues and their implementations.

Unit-2: Trees and Graphs Binary Trees and Binary Search Trees, AVL Trees and Red-Black Trees, Graph representation and traversal.

Unit-3: Hashing Introduction to hashing and its applications. Hash functions and collision resolution techniques. Hash tables and their implementation. Understanding the efficiency of hash-based data structures.

Unit-4: Sorting and Searching Algorithms Bubble Sort, Selection Sort, Insertion Sort, Merge Sort, Quick Sort, Heap Sort, Linear and Binary Search.

Unit 5: File Organization and Indexing Fundamentals of file organization: sequential, direct, and indexed. Primary and secondary indexing techniques. B+-tree index structure and its role in efficient data retrieval. Case studies of file organization in database systems.

Recommended Books:

1. "Data Structures and Algorithms in Java" by Robert Lafore.
2. "Introduction to Algorithms" by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein.

Semester II

Course: Internet of Things (SOS/CSE/MD/C203)

Course Objective:

The Internet of Things (IoT) course aims to provide students with a comprehensive understanding of the IoT ecosystem, including its architecture, design principles, and practical implementations. The course introduces core concepts such as sensing, actuation, networking, and data management, while emphasizing the role of cloud computing, security, and real-world applications. Students will gain hands-on experience with hardware platforms like Arduino and Raspberry Pi, and learn to build IoT solutions using modern communication protocols and APIs.

Course Outcome:

By the end of this course, students should be able to:

1. Understand the fundamental concepts, architecture, and applications of IoT.
2. Identify and utilize various hardware and software components involved in IoT systems.
3. Implement IoT solutions integrating sensors, actuators, and communication protocols.
4. Analyze data management strategies and security challenges in IoT.
5. Apply IoT knowledge to solve real-world problems in domains like healthcare, agriculture, and home automation.

Course Content/Syllabus:

Unit 1: Introduction to IoT

Architectural overview of IoT, Design principles and capabilities, IoT applications across industries, Introduction to sensing and actuation, Basics of networking in IoT, M2M communication, and IoT technology fundamentals.

Unit 2: Core IoT Technologies

Devices and gateways, Data management in IoT systems, Integration with business processes, Everything as a Service (XaaS), Role of cloud computing in IoT, Security aspects and best practices in IoT deployment.

Unit 3: IoT Hardware Components

Computing devices: Arduino, Raspberry Pi, Sensor and actuator technologies, Communication modules, I/O interfaces and their integration with hardware platforms.

Unit 4: IoT Software Components

Programming APIs using Python/Node.js/Arduino, Communication protocols: MQTT, ZigBee, Bluetooth, CoAP, UDP, TCP, Developing middleware for IoT data transmission and control.

Unit 5: Solution Frameworks and Case Studies

Frameworks for device integration, Data acquisition, Unstructured data storage (cloud/local), Authentication and authorization mechanisms, Case studies in industrial automation, transportation, agriculture, healthcare, and home automation.

Recommended Books:

1. Honbo Zhou, The Internet of Things in the Cloud: A Middleware Perspective, CRC Press
2. Vijay Madisetti, Arshdeep Bahga, Internet of Things: A Hands-on Approach, University Press
3. Pethuru Raj and Anupama C. Raman, The Internet of Things: Enabling Technologies, Platforms, and Use Cases, CRC Press

Semester II

Lab Manual

SOS/CS/MD/CP23: Internet of Things (IoT) Lab
Course Content: 1. Basics of HTML. 2. Basics of Web Programming (dynamic and static webpages). 3. Implementation of CSS using distinct frameworks. 4. Implementation of Javascript frameworks. 5. Deployment of Website on live server.

SOS/CS/MJ/CP21 and SOS/CS/MI/CP22: Data Structure using C Lab
Course Content: **1.** Implementation of Array ADT operations including insertion, deletion, and traversal at specific index positions. **2.** Implementation of Stack data structure and its operations (push, pop, peek) using arrays and its application in infix-to-postfix conversion. **3.** Implementation of Queue and Circular Queue data structures using arrays for enqueue and dequeue operations. **4.** Implementation of Singly Linked List and Doubly Linked List with operations like insertion, deletion, and traversal at various positions. **5.** Implementation of Stack and Queue data structures using dynamic memory allocation via Linked Lists. **6.** Implementation of Binary Search Tree (BST) operations including insertion, deletion, and tree traversals (Inorder, Preorder, Postorder). **7.** Implementation of internal sorting algorithms (Insertion Sort, Selection Sort, and Bubble Sort) along with Hashing collision resolution techniques.

Semester III

Course Code: SOS/CS/MJ/C301	**Course Title:** Object Oriented Programming in C++

Course Outcomes:

After learning the course, the students should be able to:

1. Describe the important concepts of object-oriented programming like object and class, Encapsulation, inheritance and polymorphism.
2. Write the skeleton of C++ program.
3. Write the simple C++ programs using the variables, operators, control structures, functions and I/O objects cin and cout.
4. Write the simple object-oriented programs in C++ using objects and classes.
5. Use features of C++ like type conversion, inheritance, polymorphism, I/O streams and files to develop programs for real life problems.

Course Content:

Unit-1 Introduction

Introduction to OOP, Basic Concepts of OOP, Applications of OOP. Introduction to C++, Introduction to C++ stream I/O, declarations in C++, Creating New data types in C++, function Prototypes, Inline functions, Reference Parameters, Const Qualifier, Dynamic memory allocation, default arguments, Unary Scope resolution operator, Linkage specifications.

Unit-2 Class, Constructors, Friend Class

Introduction, comparing class with Structure, Class Scope, Accessing Members of a class, Constructor, Destructor, Const objects, Const member functions, Friend class, Friend function, This pointer, Data abstraction and Information hiding, container classes and Iterators

Unit-3 Overloading & Inheritance

Operator Overloading, Fundamentals, Restrictions, Overloading stream, Insertion and stream extraction operators, overloading unary & binary operators, Converting between types, Overloading ++ and --. Inheritance, Introduction, Protected members, Casting base _class pointers to derived _class pointers Overloading Base class members in a Derived class, Public, Protocols and Private inheritance, Direct base classes and Indirect Base Classes, Using Constructors and Destructors in Derived classes, Implicit Derived class object to base class object conversion.

Unit-4 Virtual Functions

Introduction, Type fields and switch statements, Virtual functions, Abstract base classes and concrete classes, Polymorphism, Dynamic binding, Virtual destructors. C++ Stream I/O : Streams, Stream Input, Stream Output, Unformatted I/O, Stream manipulators, Stream format states, Stream error, States.

Recommended Books

S.No.	Title	Author	Publication
1.	How to Program C++	Deitel H.M. & Deitel P.J.	PHI
2.	C++ Programming	Al stevenes	Wiley dreamtech
3.	Complete Reference	Herbert Scheldt	
4.	Object Oriented Programming with C++	E. Balagurusamy	

Semester III

Course Code: SOS/CS/MI/C302	**Course Title:** Numerical Computing

Course Outcomes:

Upon completion of the course students shall be able to:

1. Recognize the error in the number generated by the solution.
2. Compute solution of algebraic and transcendental equation by numerical methods like Bisection method and Newton Raphson method.
3. Apply method of interpolation and extrapolation for prediction. Recognize elements and variable in statistics and summarize qualitative and quantitative data

Course Content:

Unit-1 Introduction

Numbers and their accuracy, Computer Arithmetic, Mathematical preliminaries, Errors and their Computation, General error formula, Error in a series approximation.

Unit-2 Solution of Algebraic and Transcendental Equation

Bisection Method, Iteration method, Method of false position, Newton-Raphson method, Methods of finding complex roots, Muller's method, Rate of convergence of Iterative methods.

Unit-3 Interpolation

Differences, Difference tables Polynomial Interpolation: Newton's forward and backward formula Central Difference Formulae: Gauss forward and backward formula, Stirling's, Bessel's, Everett's formula

Unit-4 Interpolation with Unequal Intervals

Lagrange's Interpolation, Newton Divided difference formula, Hermite's Interpolation,

Recommended Books

S.No.	**Title**	**Author**	**Publication**
1.	Computer-Based Numerical & Statistical Techniques	M. Goyal	Infinity Science Press LLC
2.	Numerical Methods	M.K. Jain	New Age
3.	Introductory Methods of Numerical Analysis	S.S. Sastry	PHI Learning

Semester III

Course Code: SOS/CS/MD/C303	**Course Title:** Introduction to Cyber Security

Course Outcomes:

After successful completion of course the student will be able to

1. Remember the broad set of technical, social & political aspects of Cyber Security.
2. Understand the importance of ethical hacking, its tool and ethical hacking process.
3. Analyze security principles to system design.
4. Understand the methods for authentication, access control, intrusion detection and prevention in Cyber Security.

Course Content

Unit-1 Introduction to Cyber Security

Importance and challenges in Cyber Security, Cyberspace, and Cyber threats, Cyber warfare, CIA Triad, Cyber Terrorism, Cyber Security of Critical Infrastructure

Unit-2 Hackers and Cyber Crimes

Types of Hackers - Hackers and Crackers, Cyber-Attacks and Vulnerabilities, Malware threats, Sniffing, Gaining Access - Escalating Privileges, Executing Applications, Hiding Files, Covering Tracks. Worms, Trojans, Viruses, Backdoors

Unit-3 Ethical Hacking

Ethical Hacking Concepts and Scopes, Threats and Attack Vectors, Information Assurance, Threat Modeling, Enterprise Information Security Architecture, Vulnerability Assessment and Penetration Testing

Unit-4 Social Engineering

Types of Social Engineering - Insider Attack - Preventing Insider Threats - Social engineering Targets and Defence Strategie

Unit -5 IT Security Act & Cyber Security Initiatives in India

IT Act, Hackers-Attacker-Countermeasures, Web Application Security, Counter Cyber Security Initiatives in India, Cyber Security Incident Handling, Cyber Security Assurance

Recommended Books

S.No.	Title	Author	Publication
1.	Cyber Security and Cyber Laws	Nilakshi Jain	Wiley
2.	Cyber Security	Nina Godbole	Wiley
3.	Beginners Guide to Ethical Hacking and Cyber Security	Abhinav Ojha	Notion Press

Semester III

Course Code: SOS/CS/SEC304	**Course Title:** Digital Electronics

Course Outcomes:

After successful completion of course the student will be able to

1. Understand number systems and logic basics used in digital electronics.
2. Apply Boolean algebra and simplification techniques to design efficient circuits.
3. Analyze different logic families based on performance parameters.
4. Design combinational circuits like adders, multiplexers, and encoders.
5. Design and analyze sequential circuits using flip-flops, registers, and counters.
6. Understand memory systems and programmable logic devices used in digital systems.

Course Content:

Unit-1 Introduction

Positional number system, binary number system, octal number system, hexadecimal number system, number system conversions, binary arithmetic, signed number representation, fixed point numbers, floating point numbers, basic logic gates, truth tables, Boolean expressions, binary codes, Gray code

Unit-2 Boolean Algebra and Switching Functions

Basic laws of Boolean algebra, Boolean theorems, representation of logic functions, SOP form, POS form, K-map simplification, Quine-McCluskey method

Unit-3 Logic Families

Diode as switch, BJT as switch, MOSFET as switch, introduction to logic families, RTL, DTL, TTL, ECL, CMOS, logic levels, noise margin, fan-out, propagation delay, power consumption, comparison of logic families

Unit-4 Combinational Logic

Adders, subtractors, arithmetic logic unit (ALU), multiplexers, demultiplexers, encoders, decoders, parity circuits, comparators

Unit -5 Sequential Logic

Latches, SR flip-flop, JK flip-flop, D flip-flop, T flip-flop, flip-flop conversions, timing hazards, races, metastability, synchronous sequential circuits, state machines, shift registers, counters, ripple counters, mod counters

Unit -6 Semiconductor Memories

RAM, ROM, content addressable memory, charge coupled devices, PLA, PAL, sequential PLDs, applications of memory devices

Recommended Books:

S.No.	Title	Author	Publication
1.	Digital Design	Morris Mano	
2.	Digital Design: Principles and Practices	John F Wakerly	Prentice Hall
3.	Digital Fundamentals	Thomas L. Floyd	Prentice Hall

Semester III

Lab Manual

Numerical Computing Lab: SOS/CS/MJ/CP32

Course Content:

1. Implementation of algebraic and transcendental equation-solving using Bisection, Regula-Falsi, and Newton-Raphson methods.
2. Implementation of system of linear equations using Gauss Elimination, Gauss-Jordan, and Gauss-Seidel iteration methods.
3. Implementation of Newton's Forward and Backward Interpolation formulae.
4. Implementation of Lagrange's Interpolation and Newton's Divided Difference Interpolation methods.
5. Implementation of numerical differentiation for first and second-order derivatives.
6. Implementation of numerical integration using Trapezoidal, Simpson's 1/3rd, and Simpson's 3/8th rules.
7. Implementation of ordinary differential equations using Euler's and Modified Euler's methods.
8. Implementation of ordinary differential equations using Runge-Kutta (RK) 2nd and 4th order methods.

OOPs using C++ Lab: SOS/CS/MJ/CP41

Course Content:

1. Implementation of basic C++ programs using classes, objects, and data hiding with access specifiers (private, public, protected).
2. Implementation of Constructor overloading, Default constructors, Parameterized constructors, Copy constructors, and Destructors.
3. Implementation of inline functions, static data members, static member functions, and Friend functions across different classes.
4. Implementation of Compile-time Polymorphism using Function Overloading and Operator Overloading (Unary and Binary operators).
5. Implementation of Run-time Polymorphism using Virtual Functions, Pure Virtual Functions, and Abstract Classes.
6. Implementation of different types of Inheritance: Single, Multiple, Hierarchical, Multilevel, and Hybrid inheritance with Virtual Base Classes.
7. Implementation of File Handling operations using ifstream, ofstream, and fstream classes for sequential and random data access.
8. Implementation of Exception Handling blocks (try, throw, catch).

Semester IV

Course Code: SOS/CS/MJ/C401	**Course Title:** Design and Analysis of Algorithms

Course Outcomes:

After successful completion of course the student will be able to

1. Understand different algorithm design techniques like divide and conquer, greedy, and dynamic programming
2. Analyze time and space complexity using asymptotic notations
3. Apply algorithms to solve real-world computational problems
4. Compare efficiency of different algorithms and choose optimal solutions
5. Understand complexity classes like P, NP, and NP-Complete problems

Course Content

Unit-1 Introduction to Algorithm + Sorting

Asymptotic analysis: Big-O, Big-Theta, Insertion Sort, Selection Sort, Bubble Sort, Heapsort, Linear Time Sorting, Selection Problem, Analysis of time complexity of all algorithms

Unit-2 Graphs

Representation of graphs, Traversal of graphs (BFS and DFS), Directed graphs, Directed Acyclic Graphs and Topological Ordering

Unit-3 Divide and Conquer

Introduction to divide and conquer technique, Merge Sort, Quick Sort with analysis of time complexity. Maximum Sub-array problem, Strassen's algorithm for matrix multiplication.

Unit-4 Greedy Algorithms

Introduction to the Greedy algorithm design approach, application to minimum spanning trees, fractional knapsack problem, shortest path problem with analysis of time complexity.

Unit -5 Dynamic Programming & Intractability

Introduction to the Dynamic Programming approach, application to weighted interval scheduling, integer knapsack problem with analysis of time complexity, Introduction to NP, NP-hard and NP-complete classes.

Recommended Books

S.No	Title	Author	Publication
1.	Introduction to Algorithms	Cormen	MIT Press
2.	Introduction to the Design and Analysis of Algorithms	Levitin	Addison Wesley
3.	Design and Analysis of Algorithms	Sandeep Sen	Cambridge University Press

Semester IV

Course Code: SOS/CS/MI/C402	**Course Title:** R Programming

Course Outcomes:

After successful completion of course the student will be able to

- Understand basics of R language and its environment
- Perform data manipulation and statistical analysis using R
- Visualize data using graphs and plots
- Write R programs for data analysis and interpretation
- Apply R in real-world applications like data science and research.

Course Content

Unit-1 Getting Started with R

Introduction to R Programming, Setting Up R Studio, Basics of R Syntax, Storing Data with Variables, Displaying Data: Input and Output

Unit-2 Working with Data

Different Data Types in R, Organizing Data: Vectors and Matrices, Cleaning and Transforming Data, Basic Data Analysis, Creating Simple Charts

Unit-3 Coding Control and Functions

Making Choices with if and else, looping with for and while, Building Own Functions, Using Functions Effectively, Handling Mistakes: Debugging

Unit-4 Creating Visual

Introduction to Data Visualization, Simple Plots with R, Customizing Graphics, Creating More Advanced Plots, Interactive Visualizations

Unit -5 Understanding Data Patterns

Exploring Data Stats, Basics of Hypothesis Testing, Understanding Regression, Introduction to Machine Learning, Practical Applications: Classification and Clustering

Recommended Books

S.No.	**Title**	**Author**	**Publication**
1.	R Programming for Data Science	Roger D. Peng	
2.	The Art of R Programming	Prashanth Singh, Vivek Mourya	Cengage Learning India

Semester IV

Course Code: SOS/CS/MD/C403	**Course Title:** E- Commerce

Course Outcomes:

After successful completion of course the student will be able to

1. Understand fundamentals and models of e-commerce
2. Analyze online business strategies and digital payment systems
3. Learn security, privacy, and legal issues in e-commerce
4. Understand working of e-commerce platforms and technologies
5. Evaluate real-world e-commerce applications and case studies

Course Content:

Unit-1 Introduction to E-Commerce

Introduction: History of E-Commerce, E-Commerce Vs Traditional Commerce, E-Commerce Terminologies and Fundamentals, E- Commerce Framework, Elements of E-Commerce Application, Benefits and Limitation of E-Commerce, m-commerce

Unit-2 Types of E-Commerce or Business Models of E-Commerce

Introduction; Basic understanding of E-Commerce Business: Models of E-Commerce Business: Business to Business (B2B): Business to Consumer (B2C): Consumer to Business (C2B): Consumer to Consumer (C2C): Business to Government (B2G); Government to Business (G2B): Business Applications of E-Commerce

Unit-3 Internet and Commerce

E-Commerce and the Internet: Benefits of E-Commerce using Internet; Issues to Internet based E-Commerce; IT Infrastructure: Internet, Intranet, Extranet; Introduction to Broadband Technology, Introduction to TCP/IP Protocol Suit: World Wide Web(www): E-mail: Communication Channels: Network Security and Firewalls;

Unit-4 Electronic Payment System

Introduction; Emergence of Electronic Payment System: Meaning of Electronic Payment System; Characteristics of E-payment System, Advantages of E-payment System: Electronic Payment Mechanism/Process: Payment Gateways: Electronic Payment Methods: Types of Payment Methods for ECommerce: Electronic payment issues; Security Properties of Electronic Payments: Electronic Payment and Website: Tools for Promoting Websites

Unit -5 Electronic Data Interchange

Basic Concepts, Type of Data to be interchanged, EDI Vs. E-mail, EDI Benefits, How EDI works, EDI Application in various field, Security and Privacy issues of EDI, EDI for E-Commerce

Recommended Books:

S.No.	Title	Author	Publication
1.	E-Commerce: Fundamentals and Applications	Henry Chan	Wiley
2.	E-Commerce	C. Kenneth	Pearson
3.	E-COMMERCE: An Indian Perspective	S. Joseph	PHI

Semester IV

Course Code: SOS/CS/SEC404	**Course Title:** Mathematical Foundation of Computer Science

Course Outcomes:

After successful completion of course the student will be able to

1. Understand fundamental mathematical concepts used in computer science
2. Apply logic, set theory, and relations in problem solving
3. Solve problems using matrices, probability, and statistics
4. Understand discrete structures like graphs and combinatorics
5. Develop analytical and logical thinking skills

Course Content

Unit-1 Sets

Sets: Set Operations, Computer Representation of Sets, Countable and Uncountable Sets, Principle of Inclusion and Exclusion, Multisets

Unit-2 Functions

Functions: One-to-one and Onto Functions, Inverse Functions and Compositions of Functions, Graphs of Functions; Sequences and Summations: Sequences, Special Integer Sequences, Summations

Unit-3 Relations

Relations: Properties of Binary Relations, Equivalence relations and Partitions, Partial Ordering Relation

Unit-4 Logic

Propositional Logic, Propositional Equivalences, Use of first-order logic to express natural language predicates, Quantifiers, Nested Quantifiers, Rules of Inference, Introduction to Proofs, Proof Methods and Strategies, Mathematical Induction

Recommended Books

S.No.	**Title**	**Author**	**Publication**
1.	Schaum's Outline of Discrete Mathematics	Seymour Lipschutz	McGraw Hill Education
2.	Introductory discrete mathematics	V. Balakrishnan	Dover
3.	Discrete Mathematics and Its Applications	Kenneth H. Rosen	McGraw Hill Education

Semester IV

Lab Manual

SOS/CS/MJ/CP41: Design and Analysis of Algorithms Lab

Course Content:

1. Implementation of sorting algorithms (Merge Sort, Quick Sort, and Heap Sort) and analyzing their time complexity.
2. Implementation of linear and binary search algorithms using Divide and Conquer approach.
3. Implementation of Minimum Spanning Tree algorithms using Kruskal's and Prim's methods.
4. Implementation of Shortest Path algorithms using Dijkstra's and Bellman-Ford algorithms.
5. Implementation of All-Pairs Shortest Path using Floyd-Warshall algorithm and Matrix Chain Multiplication using Dynamic Programming.
6. Implementation of Knapsack Problem using Greedy approach and Dynamic Programming approach.
7. Implementation of N-Queen's problem and Graph Coloring problem using Backtracking.
8. Implementation of Traveling Salesperson Problem (TSP) using Branch and Bound technique.

SOS/CS/MJ/CP42: R Programming Lab

Course Content:

1. Implementation of basic data structures in R including Scalars, Vectors, Matrices, Arrays, Data Frames, and Lists.
2. Implementation of control structures (if-else, for loops, while loops, repeat, next, and break) and writing custom user-defined functions.
3. Implementation of data importing, exporting, and manipulation operations using CSV, Excel, and text files.
4. Implementation of data cleaning and transformation techniques using dplyr, tidyr, and basic string manipulation functions.
5. Implementation of exploratory data analysis and advanced data visualization using ggplot2 (Scatter plots, Bar charts, Histograms, and Box plots).
6. Implementation of statistical analysis techniques including descriptive statistics, correlation, and covariance matrices.
7. Implementation of hypothesis testing using T-tests, ANOVA (Analysis of Variance), and Chi-Square tests.
8. Implementation of predictive modeling using Linear Regression, Logistic Regression, and time-series forecasting.

Semester V

Course Code: SOS/CS/MJ/C501	**Course Title:** Operating System

Course Outcomes:

After successful completion of course the student will be able to

1. Understand the basic concepts and functions of operating systems
2. Analyze process management, scheduling, and synchronization techniques
3. Understand memory management and virtual memory concepts
4. Learn file systems and I/O management
5. Evaluate different OS algorithms and their performance

Course Content

Unit-1 Introduction

Operating Systems (OS) definition and its purpose, Multiprogramming and Time-Sharing Systems, OS Structure, OS Operations: Dual and Multi-mode, OS as resource manager.

Unit-2 Operating System Structures

OS Services, System Calls: Process Control, File Management, Device Management, and Information Maintenance, Inter-process Communication, and Protection, System programs, OS structure- Simple, Layered, Microkernel and Modular

Unit-3 Process Management

Process Concept, States. Process Control Block, Context Switch, Process scheduling, Schedulers, Overview of threads and Scheduling Algorithms: First Come First Served, Shortest-Job-First, Priority & Round-Robin, Process Synchronization: The critical-section problem and Peterson's Solution, Deadlock characterization, Deadlock handling

Unit-4 Memory Management

Physical and Logical address space, Swapping, Contiguous memory allocation strategies - fixed and variable partitions, Segmentation, Paging, virtual memory: Demand Paging, Page Replacement algorithms: FIFO Page Replacement, Optimal Page replacement, LRU page replacement

Unit-5 File System and Disk Scheduling

File Concepts, File Attributes, File Access Methods, Directory Structure: Single-Level, Two-Level, TreeStructured, and Acyclic-Graph Directories. Mass Storage Structure: Magnetic Disks, Solid-State Disks, Magnetic Tapes, Disk Scheduling algorithms: FCFS, SSTF, SCAN, C-SCAN, LOOK, and C-LOOk Scheduling

Recommended Books

S.No.	Title	Author	Publication
1.	Operating System Concepts	Galvin	Wiley
2.	Operating Systems: A Concept-based Approach	Dhamdhere	McGraw Hill Education
3.	Operating Systems: Internals and Design Principles	W. Stallings	Pearson Education
4.	Modern Operating Systems	Andrew S. Tanenbaum	Pearson Education

Semester V

Course Code: SOS/CS/MJ/C502	**Course Title:** Computer Graphics

Course Outcomes:

After successful completion of course the student will be able to

1. Understand basic concepts of computer graphics and display systems
2. Apply transformations like translation, rotation, and scaling
3. Understand line drawing and clipping algorithms
4. Learn 2D and basic 3D graphics concepts
5. Develop simple graphical applications

Course Content

Unit-1 Introduction

Introduction to Graphics systems, Basic elements of Computer graphics, Applications of computer graphics. Architecture of Raster and Random scan display devices, input/output devices

Unit-2 Drawing and clipping primitives

Raster scan line, circle and ellipse drawing algorithms, Polygon filling, line clipping and polygon clipping algorithms

Unit-3 Transformation and Viewing

2D and 3D Geometric Transformations, 2D and 3D Viewing transformations (Projections- Parallel and Perspective), Vanishing points

Unit-4 Visible Surface determination and Surface Rendering

Z-buffer algorithm, List-priority algorithm and area subdivision algorithm for visible surface determination. Illumination and shading models, RGB Color model and Basics of Computer Animation

Recommended Books

S.No.	Title	Author	Publication
1.	Computer Graphics	Hearn, D & Baker	Prentice Hall of India
2.	Computer Graphics: Principles and Practice in C	Foley	Pearson Education
3.	Mathematical Elements for Computer Graphics	D.F. Rogers	McGraw Hill Education

Semester V

Elective II

Course Code: SOS/CS/MJ/E201	**Course Title:** Computer Networks

Course Outcomes:

After successful completion of course the student will be able to

1. Understand basic concepts and architecture of computer networks
2. Learn network protocols and communication models
3. Analyze data transmission, routing, and switching techniques
4. Understand network security and error handling
5. Evaluate performance of different networking techniques

Course Content

Unit-1 Introduction to Data Communication & Networks

Components of a data communication system, Types of networks, Network topologies and communication protocols

Unit-2 Network Architecture and OSI Model

Network architecture: client-server, peer-to-peer, OSI model layers and their functions, TCP/IP protocol suite.

Unit-3 Data Link Layer and Network Devices

Data link layer: framing, error detection, flow control, Network devices: switches, routers, bridges, etc., MAC addressing and Ethernet standards.

Unit-4 Network Security and Wireless Networks

Network security concepts: encryption, firewalls, VPN, Wireless networks and technologies

Unit-5 Network Management and Emerging Technologies

Network management and monitoring tools, Cloud computing and virtualization, Internet of Things (IoT).

Recommended Books

S.No.	**Title**	**Author**
1.	Computer Networking: A Top-Down Approach	James F. Kurose and Keith W. Ross
2.	Data Communications and Networking	Behrouz A.Forouzan and Sophia Chung Fegan

Semester V

Course Code: SOS/CS/MJ/E202	**Course Title:** Blockchain & Its Application

Course Outcomes:

After successful completion of course the student will be able to

1. Understand basic concepts of blockchain technology
2. Learn structure and working of distributed ledgers
3. Analyze cryptocurrencies and consensus mechanisms
4. Explore smart contracts and applications
5. Evaluate real-world use cases of blockchain

Course Content

Unit-1 Introduction

Evolution of digital transactions and need for blockchain Introduction to blockchain technology Features of blockchain: decentralization, transparency, immutability, traceability Blockchain vs traditional databases Structure of a block: block header, timestamp, nonce, Merkle tree, hash Chain formation and block linking Distributed Ledger Technology (DLT): concept and characteristics Public, private, and consortium blockchains Advantages and limitations of blockchain technology

Unit-2 Cryptocurrencies and Consensus Mechanisms

Introduction to cryptocurrencies Bitcoin: concept, architecture, and transaction workflow Wallets, public key and private key cryptography Digital signatures and hash functions in blockchain Mining and transaction validation Consensus mechanisms: concept and need Proof of Work (PoW) Proof of Stake (PoS) Delegated Proof of Stake (DPoS) Practical Byzantine Fault Tolerance (PBFT) Comparison of consensus algorithms

Unit-3 Blockchain 2.0 & 3.0

Introduction to smart contracts Characteristics and benefits of smart contracts Working principle of smart contracts Smart contract lifecycle Ethereum blockchain overview Ethereum Virtual Machine (EVM) Gas, accounts, and transactions in Ethereum Introduction to Solidity (basic concepts only) Writing and deploying simple smart contracts (conceptual/practical overview) Tokens and token standards: ERC-20, ERC-721 Overview of popular blockchain platforms: Ethereum, Hyperledger, Binance Smart Chain, Solana

Unit-4 Privacy & Security Issues in Blockchain

Pseudo-anonymity vs. anonymity, Zcash and Zk-SNARKS for anonymity preservation, attacks on Blockchains – such as Sybil attacks, selfish mining, 51% attacks - -advent of algorand, and Sharding based consensus algorithms to prevent these

Recommended Books

S.No.	Title	Author	Publication
1.	Blockchain Technology: Cryptocurrency and Applications	S. Shukla, M. Dhawan, S. Sharma, S. Venkatesan	Oxford University Press
2.	Mastering Blockchain	Imran Bashir	Packt Publishing
3.	Blockchain Technology Concepts and Applications	Kumar Saurabh, Ashutosh Saxena	Wiley

Semester V

Elective III

Course Code: SOS/CS/MI/E301	**Course Title:** Probability and Statistics

Course Outcomes:

After successful completion of course the student will be able to

1. Understand basic concepts of probability and statistics
2. Apply probability rules and distributions
3. Analyze data using statistical methods
4. Perform hypothesis testing and interpretation

Course Content

Unit-1

Descriptive Statistics: Types of data - qualitative and quantitative; Measures of central tendency - mean, median, mode; Measures of dispersion - range, mean deviation, variance, standard deviation; Moments, skewness and kurtosis; Frequency distributions and graphical representations: histograms, frequency polygons, ogives.

Unit-2

Probability Theory: Random experiments, sample space and events; Classical, relative frequency and axiomatic definitions; Addition and multiplication theorems; Conditional probability and Bayes' theorem; Discrete and continuous random variables; Mathematical expectation and variance; Probability distributions - Binomial, Poisson and Normal distributions and their properties.

Unit-3

Sampling Theory and Estimation: Population and sample; Methods of sampling - random, stratified, systematic and cluster; Sampling and non-sampling errors; Central Limit Theorem; Point and interval estimation; Properties of good estimators - unbiasedness, consistency, efficiency and sufficiency; Maximum likelihood and method of moments estimation.

Unit-4

Testing of Hypotheses: Null and alternative hypothesis; Type I and Type II errors; Level of significance and p-value; One-tailed and two-tailed tests; Large sample tests: z-test for mean and proportion; Small sample tests: t-test (one sample, two sample, paired); Chi-square test for goodness of fit and independence; F-test and ANOVA; Introduction to correlation and simple linear regression.

Recommended Books

S.No.	**Title**	**Author**	**Publication**
1.	Fundamentals of Mathematical Statistics	S.C. Gupta and V.K. Kapoor	Sultan Chand and Sons
2.	Introduction to Probability and Statistics	William Mendenhall, Robert J. Beaver, Barbara M. Beaver	Cengage Learning
3.	Probability and Statistics for Engineers and Scientists	Walpole, Myers and Myers	Pearson Education

Semester V

Course Code: SOS/CS/MI/E302	**Course Title:** Cyber Security and Privacy

Course Outcomes:

After successful completion of course the student will be able to

1. Understand basic concepts of cybersecurity
2. Identify common threats, attacks, and vulnerabilities
3. Learn cryptography basics and security techniques
4. Understand privacy and data protection issues
5. Apply security practices in real-world systems

Course Content

Unit-1 Fundamentals of Cyber Security and Threat Landscape

Importance and challenges in Cyber Security, Cyberspace, and Cyber threats, Cyber warfare, CIA Triad, Cyber Terrorism, Cyber Security of Critical Infrastructure

Unit-2 Cyber Attacks and Intrusion Techniques

Types of Hackers - Hackers and Crackers, Cyber-Attacks and Vulnerabilities, Malware threats, Sniffing, Gaining Access - Escalating Privileges, Executing Applications, Hiding Files, Covering Tracks. Worms, Trojans, Viruses, Backdoors

Unit-3 Ethical Hacking and Information Security Practices

Ethical Hacking Concepts and Scopes, Threats and Attack Vectors, Information Assurance, Threat Modeling, Enterprise Information Security Architecture, Vulnerability Assessment and Penetration Testing

Unit-4 Social Engineering and Insider Threats

Types of Social Engineering - Insider Attack - Preventing Insider Threats - Social engineering Targets and Defence Strategies

Unit-5 Legal Framework and Countermeasures in Cyber Security

IT Act, Hackers-Attacker-Countermeasures, Web Application Security, Counter Cyber Security Initiatives in India, Cyber Security Incident Handling, Cyber Security Assurance

Recommended Books

S.No.	Title	Author	Publication
1.	Cyber Security and Cyber Laws	Nilakshi Jain	Wiley
2.	Cyber Security	Nina Godbole	Wiley

Semester V

Elective IV

Course Code: SOS/CS/SEC/E401	**Course Title:** Programming Using C# & .NET

Course Outcomes:

After successful completion of course the student will be able to

1. Apply object-oriented programming concepts
2. Develop applications using .NET framework
3. Handle exceptions and file operations
4. Build simple GUI or web-based applications

Course Content

Unit-1 Introduction to .NET Framework and Managed Code

Introduction to .NET Framework: Managed Code and the CLR- Intermediate Language, Metadata and JIT Compilation - Automatic Memory Management

Unit-2 Language Concepts, CLR, and Framework Class Library

Language Concepts and the CLR: Visual Studio .NET - Using the .NET Framework. The Framework Class Library: NET objects - ASP .NET - .NET web services – Windows Forms

Unit-3 ASP.NET Features and Web Services

ASP.NET Features: Change the Home Directory in IIS - Add a Virtual Directory in IIS- Set a Default Document for IIS - Change Log File Properties for IIS - Stop, Start, or Pause a Web Site.

Unit-4 Web Controls and Creating Web Forms

Creating Web Controls: Web Controls - HTML Controls, Using Intrinsic Controls, Using Input Validation Controls, Selecting Controls for Applications - Adding web controls to a Page. Creating Web Forms: Server Controls - Types of Server Controls - Adding ASP.NET Code to a Page

Recommended Books

S.No.	**Title**	**Author**
1.	Professional ASP.NET 4.5 in C# and VB	Jason N. Gaylord, Christian Wenz, Pranav Rastogi, Todd Miranda, and Scott Hanselman
2.	ASP.NET Core 5 for Beginners	Jonas Fagerberg

Semester V

Course Code: SOS/CS/SEC/E402	**Course Title:** Wireless & Mobile Computing

Course Outcomes:

After successful completion of course the student will be able to

1. Understand wireless communication basics
2. Learn mobile computing architecture and protocols
3. Analyze wireless network standards and technologies
4. Understand mobility management and security
5. Evaluate applications of mobile computing

Course Content

Unit-1 Existing Wireless Systems

Introduction, Global System for Mobile Communications (GSM), Personal Communications Services (PCS), International Mobile Telecommunications 2000 (IMT-2000). Wireless local area networks. Wireless local loops.

Unit-2 Wireless and Mobile Computing

Overview of the history, evolution, and compatibility of wireless standards; the special problems of wireless and mobile computing.

Unit-3 Mobile Communication System

Introduction, mobile Internet protocol; mobile aware adaptation; extending the client-server model to accommodate mobility; mobile data access; the software packages to support mobile and wireless computing; Mobile/Cellular System Infrastructure, Registration, Handoff Parameters and Underlying Support. Roaming Support Using System Backbone. The role of middleware

Unit-4 Satellite Systems

Introduction, Types of Satellite Systems, Characteristics of Satellite Systems, Satellite System Infrastructures, Call Setup, Global Positioning System.

Unit-5 Recent Advances

Introduction, Ultra-Wideband Technology, Multimedia Services Requirements. Mobility Management for Integrated Systems; Multicast in Wireless Networks; MANET Route Maintenance/Repair; Design Issues in Sensor Networks; Bluetooth Networks; Threats and Security Issues.

Recommended Books

S.No.	**Title**	**Author**
1.	Introduction to Wireless and Mobile Systems	Dharma P. Agrawa
2.	Internetworking with TCP/IP Vol.1: Principles, Protocols, and Architecture	Comer, Douglas E.

Semester V

Basic Unix Lab: SOS/CS/MJ/CP51

Course Content

1. Demonstrate how to use the following Bourne Shell commands: cat, grep, ls, more, ps, chmod, finger etc.
2. Use the following Bourne Shell constructs: test, if then, if then else, if then el if, for, while, until, and case.
3. Copy, move, and delete files and directories.
4. Write simple Shell scripts.
5. Make a Shell script executable
6. Create a ".profile" script to customize the user environment..
7. Execute programs written in C under UNIX environment

Computer Graphics Lab: SOS/CS/MJ/CP52

Course Content

1. Implementation of line generation using slope's method, DDA and Bresenham algorithms.
2. Implementation of circle generation using Mid-point method and Bresenham algorithm
3. Implementation of ellipse generation using Mid-point method.
4. Implementation of polygon filling using Flood-fill, Boundary-fill and Scan- line algorithms.
5. Translation, Scaling, Rotation, Mirror Reflection and Shearing.
6. 3D Translation, Scaling and rotation.
7. Implementation of Line Clipping using Cohen-Sutherland algorithm and Bisection Method.
8. Implementation of Polygon Clipping using Sutherland-Hodgman algorithm.

Semester VI

Course Code: SOS/CS/MJ/C601	**Course Title:** Basic Java Programming

Course Outcomes:

After successful completion of course the student will be able to

1. Understand fundamentals of Java programming language
2. Apply object-oriented concepts like classes, objects, inheritance, and polymorphism
3. Develop programs using control statements and functions
4. Handle exceptions and basic file operations
5. Build simple Java-based applications

Course Content

Unit-1 Introductory Concepts

Program, identifiers, variables, constants, primitive datatypes, expressions, Naming Conventions, Type casting, operators, control statements, structured data types, arrays, functions.

Unit-2 Object Oriented Concepts

Abstraction, encapsulation, objects, classes, methods, constructors, inheritance, polymorphism, static and dynamic binding, Anonymous block, Static Data members, overloading and overriding, Usage of super and this keyword, Abstract classes, Interfaces and Packages, Access modifiers, object class.

Unit-3 Database Programming

Database Programming, The Design of JDBC. The Structured Query Language, JDBC Installation, Basic JDBC Programming Concepts, Query Execution, Scrollable and Updatable Result Sets, Metadata, Row Sets, Transactions.

Unit-4 AWT

AWT: Components and container used in AWT, Layout managers, Listeners and Adapter classes, Event Delegation model, Swing: Introduction to Swing Component and Container Classes

Recommended Books

S.No.	Title	Author	Publication
1.	The Complete Reference	Naughton, Schildt	TMH
2.	Programming in JAVA	Balagurusamy E	TMH
3.	Java2 Black book	Steven Holzner	Dreamtech

Semester VI

Course Code: SOS/CS/MJ/C602	**Course Title:** Database Management System

Course Outcomes:

After successful completion of course the student will be able to

1. Understand database concepts and architecture
2. Design databases using ER models and relational models
3. Apply SQL for data manipulation and querying
4. Understand normalization and transaction management
5. Learn concepts of indexing, concurrency, and recovery

Course Content

Unit-1 Introduction to Database

Introduction to database and DBMS, architecture of DBMS, data independence, components of database systems, difference between traditional file system and database management system

Unit-2 E-R Model

Entity types, types of attributes and key, relationships types, constraints on relationship, modeling ER diagram, Database design using ER diagrams

Unit-3 Relational Data Model

Relational model concepts, relation characteristics, relational integrity constraints, types of keys, update anomalies

Unit-4 Normalization

Functional dependencies, Use of normalization, First, Second and Third normal forms, BCNF

Unit-5 SQL

Overview of SQL query language, Data definition and manipulation languages, set operations. DDL: to construct the database and alter its contents, DML: Querying in SQL to retrieve data from the database, use of join and simple aggregate functions

Recommended Books

S.No.	**Title**	**Author**	**Publication**
1.	Database System Concepts	S. Abraham	McGraw Hill Education
2.	Introduction to Database Management Systems	Atul Kahate	Pearson
3.	Learning SQL	Alan Beaulieu	O'Reilly

Semester VI

Elective V

Course Code: SOS/CS/MJ/E501	**Course Title:** Discrete Mathematics

Course Outcomes:

After successful completion of course the student will be able to

1. Understand logic, sets, and relations
2. Apply combinatorics and graph theory
3. Solve problems using mathematical reasoning
4. Understand functions and algebraic structures
5. Develop analytical and problem-solving skills

Course Content

Unit-1 Mathematical Logic

Propositional Calculus: Statements and Notations, Connectives, Well Formed Formulas, Truth Tables, Tautologies, Equivalence of Formulas, Duality Law, Tautological Implications, Normal Forms, Predicates, Predicative Logic, Statement Functions, Variables and Quantifiers

Unit-2 Set Theory

Sets: Operations on Sets, Principle of Inclusion-Exclusion, Relations: Properties, Operations, Partition and Covering, Transitive Closure, Equivalence, Compatibility and Partial Ordering, Hasse Diagrams

Unit-3 Functions

Bijective, Composition, Inverse, Permutation, and Recursive Functions, Lattice and its Properties, Algebraic Structures: Algebraic Systems, Properties, Semi Groups and Monoids, Group, Subgroup and Abelian Group, Homomorphism, Isomorphism

Unit-4 Combinatorics

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

Unit-5 Number Theory

Properties of Integers, Division Theorem, Greatest Common Divisor, Euclidean Algorithm, Least Common Multiple, Testing for Prime Numbers, The Fundamental Theorem of Arithmetic, Modular Arithmetic, Fermat's and Euler's Theorems

Recommended Books

S.No.	Title	Author	Publication
1.	Discrete Mathematical Structures with Applications to Computer Science	J. P. Tremblay and P. Manohar	Tata McGraw Hill
2.	Elements of Discrete Mathematics-A Computer Oriented Approach	C. L. Liu and D. P. Mohapatra	Tata McGraw Hill

Semester VI

Course Code: SOS/CS/MJ/E502	**Course Title:** Introduction to Data Science

Course Outcomes:

After successful completion of course the student will be able to

1. Understand data science concepts and workflow
2. Perform data collection and preprocessing
3. Apply basic statistical and analytical techniques
4. Visualize and interpret data
5. Use tools for data-driven decision making

Course Content

Unit-1

Introduction to Data Science: Definition and overview of data science; The data science process - data collection, preparation, exploration, modeling and communication; Data science roles and skills; Types of data - structured, semi-structured and unstructured; Introduction to Python for data science: NumPy and Pandas basics; Data acquisition methods - web scraping, APIs, databases and file formats (CSV, JSON, XML).

Unit-2

Data Wrangling and Exploration: Data cleaning - handling missing values, duplicates and outliers; Data transformation - normalization, standardization, encoding categorical variables; Exploratory Data Analysis (EDA): descriptive statistics and correlation analysis; Data visualization using Matplotlib and Seaborn: line plots, scatter plots, bar charts, histograms, box plots and heatmaps; Feature selection and feature engineering techniques.

Unit-3

Statistical Foundations and Machine Learning Basics: Probability and statistics review for data science; Supervised vs. unsupervised learning; Regression: simple and multiple linear regression, logistic regression; Classification algorithms: k-Nearest Neighbors, Decision Trees and Naive Bayes; Model evaluation metrics: accuracy, precision, recall, F1-score and RMSE; Train-test split and cross-validation.

Unit-4

Advanced Concepts and Applications: Clustering: k-Means and hierarchical clustering; Dimensionality reduction: Principal Component Analysis (PCA); Introduction to Big Data: Hadoop ecosystem and MapReduce concepts; SQL for data science: querying and aggregating data; Real-world applications in healthcare, finance, e-commerce and social media analytics; Data science project workflow: problem formulation, hypothesis testing, model deployment and storytelling with data; Ethics and privacy in data science.

Recommended Books

S.No.	Title	Author	Publication
1.	Data Science from Scratch	Joel Grus	O'Reilly Media
2.	Python for Data Analysis	Wes McKinney	O'Reilly Media
3.	Introduction to Data Science: A Python Approach to Concepts, Techniques and Applications	Laura Igual, Santi Segui	Springer

Semester VI

Elective VI

Course Code: SOS/CS/MI/E601	**Course Title:** Artificial Intelligence

Course Outcomes:

After successful completion of course the student will be able to

1. Understand basic concepts of AI and intelligent systems
2. Learn search algorithms and problem-solving techniques
3. Apply knowledge representation methods
4. Understand machine learning basics
5. Explore real-world AI applications

Course Content

Unit-1 Introduction to Artificial Intelligence

Overview of artificial intelligence and its applications, History and foundations of AI, Ethical considerations in AI development and deployment

Unit-2 Problem Solving and Search Algorithms

Problem-solving techniques and algorithms, Search algorithms: uninformed and informed search, Heuristic search and optimization algorithms

Unit-3 Knowledge Representation and Reasoning

Knowledge representation techniques: logic, semantic networks, and frames, Inference mechanisms and reasoning algorithms, Uncertainty and probabilistic reasoning in AI

Unit-4 Machine Learning and AI Models

Introduction to machine learning algorithms, Supervised, unsupervised, and reinforcement learning, Deep learning and neural networks for AI applications

Unit-5 AI Systems and Applications

Natural language processing and understanding, Computer vision and image recognition, AI in robotics and autonomous systems

Recommended Books

S.No.	**Title**	**Author**	**Publication**
1.	Artificial Intelligence: A Modern Approach	Stuart Russell and Peter Norvig	
2.	Introduction to Artificial Intelligence	Akash Kaushik	Cengage

Semester VI

Course Code: SOS/CS/MI/E602	**Course Title:** Distributed System

Course Outcomes:

After successful completion of course the student will be able to

1. Understand concepts of distributed computing systems
2. Learn communication and coordination between systems
3. Analyze synchronization and consistency issues
4. Understand fault tolerance and reliability
5. Evaluate real-world distributed applications

Course Content

Unit-1

Introduction to Distributed Systems: Definition, goals and characteristics; Models: client-server, peer-to-peer and multi-tier architectures; Examples: Internet, intranets and network file systems; Types: distributed computing, information and pervasive systems; Inter-process communication: remote procedure calls (RPC), remote method invocation (RMI), message-oriented communication and message passing.

Unit-2

Synchronization and Coordination: Clock synchronization: physical clocks and logical clocks (Lamport timestamps, vector clocks); Mutual exclusion: centralized, token ring and Ricart-Agrawala algorithms; Election algorithms: bully and ring algorithms; Distributed transactions: ACID properties; Concurrency control: two-phase locking; Deadlock detection and resolution in distributed systems.

Unit-3

Replication, Consistency and Naming: Replication: reasons and data-centric consistency models (strict, sequential, causal); Client-centric consistency models; Replica management and update propagation; Naming: flat, structured and attribute-based naming; Name resolution and DNS; Distributed file systems: NFS and AFS; Directory services and LDAP.

Unit-4

Fault Tolerance and Security: Dependability - availability, reliability, safety and maintainability; Process resilience and failure models; Reliable group communication; Distributed commit: two-phase and three-phase commit protocols; Recovery: checkpointing and logging; Security: secure channels, authentication and access control; Introduction to web services (SOA, REST); Case studies: Google File System (GFS), Apache Hadoop and Apache Kafka.

Recommended Books

S.No.	Title	Author	Publication
1.	Distributed Systems: Principles and Paradigms	Andrew S. Tanenbaum and Maarten Van Steen	Pearson Education
2.	Designing Distributed Systems	Brendan Burns	O'Reilly Media
3.	Distributed Systems: Concepts and Design	George Coulouris, Jean Dollimore, Tim Kindberg, Gordon Blair	Pearson Education

Semester VI

Lab Manual

SOS/CS/MJ/CP61: Basic Java Programming Lab

Course Content:

1. Creation of classes and use of different types of functions
2. Programs using constructor and destructor
3. Programs on interfaces
4. Programs on packages
5. Programs using function overloading
6. Programs using inheritance
7. Programs using IO streams

SOS/CS/MJ/CP62: Database Management System Lab

Course Content:

1. Write the queries for Data Definition and Data Manipulation language.
2. Write SQL queries using Logical operators (=, <,>, etc.).
3. Write SQL queries using SQL operators (Between…. AND, IN(List), Like, ISNULL and also with negating expressions).
4. Write SQL query using character, number, date and group functions.
5. Write SQL queries for Relational Algebra (UNION, INTERSECT, and MINUS, etc.).
6. Write SQL queries for extracting data from more than one table (Equi-Join, Non-Equi-Join, Outer Join)
7. Write SQL queries for sub queries, nested queries. 8. Write programs by the use of PL/SQL.

Semester VII (UG with Honours & UG Honours with Research*)

SOS/CSE/MJ/C701: Computer Organization and Architecture

Course Objective:

This course aims to provide a detailed understanding of the fundamental concepts of computer organization and architecture. Students will explore how computers represent data, perform computations, manage memory, and communicate with peripheral devices. The course includes topics such as logic circuits, hardware structure, instruction execution, input/output operations, and memory management. Through theoretical concepts and practical design exposure, learners will gain insight into the internal workings of modern computing systems.

Course Content/Syllabus:

Unit 1: Representation of Information and Basic Building Blocks

Number systems: Binary, Octal, Hexadecimal and their conversions; Character codes: BCD, ASCII, EBCDIC; Digital codes: Gray Code, XS-3 Code.

Unit 2: Logic Circuits

Basic logic functions; Synthesis of logic using AND, OR, NOT gates; Minimization of logic expressions; Synthesis using NAND and NOR gates; Implementation of logic gates; Flip-Flops, Registers and Shift Registers, Counters, Decoders, Multiplexers, Programmable Logic Devices; Sequential circuits.

Unit 3: Basic Structure of Computer Hardware and Software

Functional units of a computer; Basic operational concepts; Bus structures; Introduction to software; Performance evaluation; Concepts of distributed computing.

Basic concepts of memory locations and operations; Addressing modes; Basic I/O operations; Data structures: stacks and queues; Subroutine implementation and use.

Unit 4: Processing Unit

Fundamental concepts of instruction execution; Hardwired vs. Microprogrammed control; Performance considerations; Arithmetic operations (signed addition/subtraction, multiplication, division); Branching conditions; Floating-point numbers and arithmetic operations.

Unit 5: Input-Output Organization

Accessing I/O devices; Interrupt handling; Direct Memory Access (DMA); I/O hardware; Standard I/O interfaces.

Unit 6: Memory Organization

Semiconductor RAM and ROM; Cache memory and performance considerations; Virtual memory concepts; Memory management requirements and techniques.

Introduction to Computer Peripherals Overview of input/output devices; On-line storage concepts and devices.

References:

1. Willam Stalling, "Computer Organization and Architecture" Pearson Education Asia
2. Mano Morris, "Computer System Architecture" PHI
3. Zaky and Hamacher, "Computer Organization: McGraw Hill
4. B. Ram, "Computer Fundamental Architecture and Organization" New Age

***Note: In cases where a subject appears in both the UG Honours and UG Honours with Research programme structures, the syllabus prescribed for that subject shall be the same.**

Semester VII (UG with Honours & UG Honours with Research*)

Course: Theory of Computation (SOS/CSE/MJ/C702)

Course Objective:

This course introduces students to the foundational concepts of computation theory. It covers mathematical models of computation such as finite automata, pushdown automata, and Turing machines. The course emphasizes the classification of formal languages, grammar simplification, and decision algorithms. Through this, students develop a rigorous understanding of what problems can be solved computationally and the limits of computing machines.

Course Content/Syllabus:

Unit 1 - Introduction to the Theory of Computation and Finite Automata: Mathematical preliminaries and Notation, three basic concepts, applications, deterministic Finite Acceptors, Nondeterministic finite acceptors, equivalence of Deterministic and Nondeterministic finite acceptors, reduction of the Number of states in Finite Automata.

Unit 2 - Regular Languages, Regular Grammars and Properties of Regular Languages: Regular expressions, connection between regular expressions and regular languages, regular grammars, closure properties of regular languages, elementary questions about regular languages, identifying language.

Unit 3 - Context-Free Languages and Simplification of Context-Free Grammars and Normal Forms: Context-free grammars, parsing and ambiguity, context-free grammars and programming languages, methods of transforming grammars, two important normal forms.

Unit 4 - Pushdown Automata and Properties of Context-Free Languages: Nondeterministic pushdown automata, pushdown automata and context-free language, deterministic pushdown automata and deterministic context-free languages, two pumping lemmas, closure properties and decision algorithms for context-free language.

Unit 5 - Turing Machines and Other Models of Turing Machines: The standard Turing machine, combining Turing machines for complicated tasks, Turing's thesis, a minor variation on the Turing machine, combining Turing machines, a universal Turing machine.

Recommended Books:

1. An Introduction to Formal Languages and Automata – Peter Linz, Narosa Publishing House, 1997.
2. Introduction to Languages and the Theory of Automata – John C. Martin, McGraw-Hill, 1997.
3. Introduction to Automata Theory, Languages, and Computation – J.P. Hopcroft, J.D. Ullman, Narosa Publishing.

***Note: In cases where a subject appears in both the UG Honours and UG Honours with Research programme structures, the syllabus prescribed for that subject shall be the same.**

Semester VII (UG with Honours & UG Honours with Research*)

Course: Advanced Operating System (SOS/CSE/MJ/C703)

Course Objective:

This course is designed to provide students with an in-depth understanding of the design, implementation, and functioning of advanced operating systems. It covers memory management, CPU scheduling, process synchronization, deadlocks, device and file system management, as well as the structure and characteristics of various types of operating systems such as batch systems, time-sharing systems, distributed systems, and real-time systems. Through this course, students will develop a thorough understanding of how an OS manages system resources and ensures efficient execution of programs.

Course Content/Syllabus:

Unit 1 - Introduction: What is an Operating System, Simple Batch Systems, Multiprogrammed Batch Systems, Time-Sharing Systems, Personal Computer Systems, Parallel Systems, Distributed Systems, Real-Time Systems.

Unit 2 - Memory Management: Background, Logical versus Physical Address Space, Swapping, Contiguous Allocation, Paging, Segmentation, Segmentation with Paging. Virtual Memory: Demand Paging, Page Replacement, Page Replacement Algorithms, Performance of Demand Paging, Allocation of Frames, Thrashing, Other Considerations, Demand Segmentation.

Unit 3 - Processes: Process Concept, Process Scheduling, Operation on Processes, Cooperating Processes, Interprocess Communication. CPU Scheduling: Basic Concepts, Scheduling Criteria, Scheduling Algorithms, Multiple-Processor Scheduling, Real-Time Scheduling, Algorithm Evaluation. Process Synchronization: Background, The Critical-Section Problem, Synchronization Hardware, Semaphores, Classical Problems of Synchronization, Critical Regions, Monitors, Synchronization in Solaris 2, Atomic Transactions. Deadlocks: System Model, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlock, Combined Approach to Deadlock Handling.

Unit 4 - Device Management: Techniques for Device Management, Dedicated Devices, Shared Devices, Virtual Devices; Device Characteristics – Hardware Consideration, Input or Output Devices, Storage Devices, Channels and Control Units, Independent Device Operation, Buffering, Multiple Paths, Block Multiplexing, Device Allocation Consideration. Secondary Storage Structure: Disk Structure, Disk Scheduling, Disk Management, Swap-Space Management, Disk Reliability, Stable-Storage Implementation.

Unit 5 - Information Management: Introduction, A Simple File System, General Model of a File System, Symbolic File System, Basic File System, Access Control Verification, Logical File System, Physical File System. File-System Interface: File Concept, Access Methods, Directory Structure, Protection, Consistency Semantics. File-System Implementation: File-System Structure, Allocation Methods, Free-Space Management, Directory Implementation, Efficiency and Performance, Recovery.

Recommended Books:

1. Abraham Silberschatz and Peter Baer Galvin, Operating System Concepts
2. Milan Milankovic, Operating Systems: Concepts and Design, McGraw-Hill
3. R. C. Joshi, Operating System, Wiley Dreamtech India Pvt. Ltd.
4. Harvey M. Deitel, Operating Systems, Addison-Wesley

***Note: In cases where a subject appears in both the UG Honours and UG Honours with Research programme structures, the syllabus prescribed for that subject shall be the same.**

Semester VII (UG with Honours & UG Honours with Research*)

Elective VII

Machine Learning: SOS/CS/MJ/E701

Course Objective:

This course aims to introduce students to the foundational concepts and techniques of machine learning. It covers various learning paradigms, including supervised, unsupervised, and reinforcement learning, and dives into core algorithms such as decision trees, Bayesian models, neural networks, genetic algorithms, and clustering techniques. The course is designed to equip students with both theoretical knowledge and practical skills to design, implement, and evaluate machine learning systems for real-world problems.

Course Content:

Unit 1 - Introduction: Well-Posed learning problems, Basic concepts, Designing a learning system, Issues in machine learning. Types of machine learning: Learning associations, Supervised learning, Unsupervised learning, Reinforcement learning

Unit 2 - Decision Tree Learning: Decision tree representation, appropriate problems for decision tree learning, Univariate Trees (Classification and Regression), Multivariate Trees, Basic Decision Tree Learning algorithms, Hypothesis space search in decision tree learning, Inductiv bias in decision tree learning, Issues in decision tree learning.

Unit 3 - Bayesian Learning: Bayes theorem and concept learning, Bayes optimal classifier, Gibbs algorithms, Naive Bayes Classifier, Bayesian belief networks, The EM algorithm. Artificial Neural Network: Neural network representation, Neural Networks as a paradigm for parallel processing, Linear discrimination, pairwise separation, Gradient Descent, Logistic discrimination, Perceptron, Training a perceptron, Multilayer perceptron, Back propagation Algorithm. Recurrent Networks, dynamically modifying network structure.

Unit 4 - Genetic Algorithms: Basic concepts, Hypothesis space search, Genetic programming, Models of evolution and learning, Parallelizing Genetic Algorithms.

Unit 5 - Data Mining Techniques for Analysis: Classification: Decision tree induction, Bayes classification, Rule-based classification, Support Vector Machines, Classification Using Frequent Patterns, k-Nearest-Neighbor, Fuzzy-set approach Classifier, Clustering: K-Means, k-Medoids, Agglomerative versus Divisive Hierarchical Clustering Distance Measures in Algorithmic Methods, Mean-shift Clustering.

References:

1. Mitchell T.M., Machine Learning, McGraw Hill
2. Bishop C., Pattern Recognition and Machine Learning, Springer Verlag.
3. Stephen Marsland, Machine Learning: An Algorithmic Perspective, CRC Press

***Note: In cases where a subject appears in both the UG Honours and UG Honours with Research programme structures, the syllabus prescribed for that subject shall be the same.**

Semester VII (UG with Honours & UG Honours with Research*)

Cloud Computing: SOS/CS/MJ/E702

Course Objectives

1. To understand the various distributed system models and evolving computing paradigm.
2. To gain knowledge in virtualization of computer resources.
3. To realize the reasons for migrating into cloud.
4. To introduce the various levels of services that can be achieved by a cloud.
5. To describe the security aspects in cloud and the services offered by a cloud.

Course Content:

UNIT - I

Cloud Computing Fundamentals: Definition of Cloud computing, Roots of Cloud Computing, Layers and Types of Clouds, Desired Features of a Cloud, Cloud Infrastructure Management, Infrastructure as a Service Providers, Platform as a Service Providers. Computing Paradigms: High Performance Computing, Parallel Computing, Distributed Computing, Cluster Computing, Grid Computing.

UNIT- II

Migrating into a Cloud: Introduction, Broad Approaches to Migrating into the Cloud, the Seven Step Model of Migration into a Cloud. Virtualization: Virtual Machines and Virtualization of Clusters and data centers - Implementation Levels of Virtualization -Virtualization Structures/Tools and Mechanisms- Virtualization of CPU, Memory, and I/O Devices-Virtual Clusters and Data Centers.

UNIT- III

Infrastructure as a Service (IAAS) & Platform (PAAS): Virtual machines provisioning and Migration services, Virtual Machines Provisioning and Manageability, Virtual Machine Migration Services, VM Provisioning and Migration in Action. On the Management of Virtual machines for Cloud Infrastructures- Aneka—Integration of Private and Public Clouds.

UNIT- IV

Software as a Service (SAAS) &Data Security in the Cloud: Software as a Service SAAS), Google App Engine – Centralizing Email Communications- Collaborating via Web Based Communication Tools-An Introduction to the idea of Data Security.

UNIT- V

SLA Management in cloud computing: Traditional Approaches to SLO Management, Types of SLA, Life Cycle of SLA, SLA Management in Cloud.

References:

1. Rajkumar Buyya, Christian Vecchiola, Thamarai Selvi, Mastering Cloud Computing: Foundations and Applications Programming, McGraw Hill Education.
2. Anthony T. Velte, Toby J. Velte, Robert Elsenpeter, Cloud Computing: A Practical Approach, McGraw Hill Education.

***Note: In cases where a subject appears in both the UG Honours and UG Honours with Research programme structures, the syllabus prescribed for that subject shall be the same.**

Semester VII (UG with Honours & UG Honours with Research*)

SOS/CSE/MI/C704: Natural Language Processing

Course Objective:

This course aims to introduce students to the fundamental concepts, models, and algorithms used in natural language processing (NLP). The course covers core areas including morphological analysis, syntactic and semantic processing, and language generation. Students will learn how natural language is processed computationally using finite-state automata, grammar rules, probabilistic models, and machine learning techniques. The course also introduces practical applications such as machine translation, information retrieval, and conversational agents.

Course Content:

Unit 1 - Regular expressions and automata, Morphology and Finite State transducers, N – grams.

Unit 2 - Word classes and part of speech tagging, Context free grammars for English, Parsing with context free grammars.

Unit 3 - Features and Unifications, Lexicalized and Probabilistic parsing.

Unit 4 - Semantics: Representing meaning, Semantic analysis, Lexical semantics, Word Scene Disambiguation and Information retrieval.

Unit 5 - Pragmatics: Discourse, Dialog and Conversational Agents, Natural Language Generation, Machine Translation.

References:

1. Daniel, Jurafsky and Martin, Speech and Language Processing, Pearson, 2003

***Note: In cases where a subject appears in both the UG Honours and UG Honours with Research programme structures, the syllabus prescribed for that subject shall be the same.**

Semester VII (UG with Honours & UG Honours with Research*)

Lab Manual

Shell Programming Lab: SOS/CS/MJ/CP705

Course Content:

1. Implementation of foundational Linux terminal navigation commands, file permission settings (chmod, chown), and directory management.
2. Implementation of text processing filters and pipeline configurations using grep, sed, awk, cut, and sort utilities.
3. Implementation of basic shell scripts using environment variables, positional parameters, and user-input processing routines.
4. Implementation of conditional logic branches in shell scripts using if-then-else, nested blocks, and case structures.
5. Implementation of shell looping constructs using for, while, and until loops for text file scanning and utility automation.
6. Implementation of shell arithmetic operations and customized string manipulation routines using expr and system utilities.
7. Implementation of robust process monitoring scripts using signals, background execution tools (&, nohup), and system cron jobs (crontab).
8. Implementation of administrator utility scripts for system backup compression (tar), disk space tracking, and network connectivity automation.

Digital Electronics Lab: SOS/CS/MJ/CP42

Course Content:

1. Implementation and verification of truth tables for basic logic gates (AND, OR, NOT) and universal gates (NAND, NOR).
2. Implementation of Boolean expressions and logic simplification using K-Maps and realization via NAND/NOR universal gates.
3. Implementation of combinational arithmetic circuits including Half Adder, Full Adder, Half Subtractor, and Full Subtractor.
4. Implementation of data routing combinational circuits including Multiplexers (MUX), Demultiplexers (DEMUX), Encoders, and Decoders.
5. Implementation and truth table verification of sequential circuits including SR, JK, D, and T Flip-Flops.
6. Implementation of shift registers including Serial-In Parallel-Out (SIPO) and Parallel-In Serial-Out (PISO) configurations.

***Note: In cases where a subject appears in both the UG Honours and UG Honours with Research programme structures, the syllabus prescribed for that subject shall be the same.**

Semester VIII (UG with Honours & UG Honours with Research*)

Compiler Design: SOS/CSE/MJ/801

Course Objectives:

1. To understand the structure and functionality of a compiler.
2. To study the various phases involved in the compilation process.
3. To learn lexical analysis, syntax analysis, semantic analysis, intermediate code generation, and code optimization techniques.
4. To explore the use of tools like LEX and YACC for compiler construction.
5. To develop the ability to design and implement a simple compiler.

Course Content:

Unit 1 - Compiler Structure: Compilers and Translators, Various Phases of Compiler, Pass Structure of Compiler, Bootstrapping of Compiler. Programming Language: High level languages, lexical and syntactic structure of a language, Data elements, Data Structure, Operations, Assignments, Program unit, Data Environments, Parameter Transmission.

Unit 2 - Lexical Analysis: The role of Lexical Analyzer, A Simple approach to the design of Lexical Analyzer, Regular Expressions, Transition Diagrams, Finite state Machines, Implementation of Lexical Analyzer, Lexical Analyzer Generator: LEX, Capabilities of Lexical Analyzer.

Unit 3 - The Syntactic Specification of Programming Languages: CFG, Derivation and Parse tree, Ambiguity, Capabilities of CFG. Basic Parsing Techniques: Top-Down parsers with backtracking, Recursive descent Parsers, Predictive Parser, Bottom-up Parsers, Shift-Reduce Parsing, Operator Precedence Parsers, LR parsers (SLR, Canonical LR, LALR) Syntax Analyzer Generator: YACC

Unit 4 - Intermediate Code Generation: Different Intermediate forms: Three address code, Quadruples and Triples, Syntax Directed Translation mechanism and attributed definition. Translation of Declaration, Assignment, Control flow, Boolean expression, Array References in arithmetic expressions, procedure calls, case statements, postfix translation. Run Time Memory Management: Static and Dynamic storage allocation, stack-based memory allocation schemes, Symbol Table management. Error Detection and Recovery: Lexical phase errors. Syntactic phase errors, semantic errors.

Unit 5 - Code Optimization and Code Generation: Local optimization, Peephole optimization, Basic blocks and flow Graphs, DAG, Data flow analyzer, Machine Model, Order of evaluation, Register allocation and code selection.

References:

1. Alfred V Aho, Jeffrey D. Ullman, "Principles of Compiler Design", Narosa
2. A.V. Aho, R. Sethi and J.D.Ullman, "Compiler Principle, Tech and tools"AW
3. H.C. Holub "Compiler Design in C", Printice Hall Inc.
4. Apple, "Modern Computer Implementation in C: Basic Design" Cambridge Press
5. Modern Compiler Design: Dick Grune, Wiley dreamtech India Pvt. Ltd.
6. “Starting Out with Modern Compiler” David Gaddis Wiley dreamtech India Pvt. Ltd.

***Note: In cases where a subject appears in both the UG Honours and UG Honours with Research programme structures, the syllabus prescribed for that subject shall be the same.**

Semester VIII (UG with Honours & UG Honours with Research*)

Network Security and Cryptography: SOS/CSE/MJ/C802

Course Objectives:

1. To introduce the fundamental concepts of cryptography and network security.
2. To explore various encryption techniques, including both classical and modern ciphers.
3. To understand public key infrastructure, digital signatures, and key management.
4. To study cryptographic hash functions and message authentication codes.
5. To understand authentication protocols, web and email security mechanisms.

Course Content:

Unit 1 - Introduction of Cryptography: Introduction To security: Attacks, Services and Mechanisms, Security, Attacks, Security Services, Conventional Encryption: Classical Techniques, Conventional Encryption Model, and steganography, Classical Encryption Techniques. Modern Techniques: Simplified DES, Block Cipher Principles, DES Standard, DES Strength, Differential and Linear Cryptanalysis, Block Cipher Design Principles, Block Cipher Modes of Operations.

Unit 2 - Conventional Encryption Algorithms: Triples DES, Blowfish, International Data Encryption Algorithm, RCS, CAST-128, CR2 Placement and Encryption Function, Key Distribution, Random Number Generation, Placement of Encryption Function.

Unit 3 - Public Key Encryption: Public-Key Cryptography: Principles of Public-Key Cryptosystems, RSA Algorithm, Key, Key Management, Fermat's and Euler's Theorm, Primality, Chinese Remainder Theorem. Hash Functions: Message Authentication and Hash Functions: Authentication Requirements, Authentication Functions, Message Authentication Codes, Hash Function Birthday Attacks, Security of Hash Function and MACS, MD5 Message Digest Algorithm, Secure Hash Algorithm (SHA), Digital Signatures: Digital Signature, Authentication Protocol, Digital Signature Standard (DDS) Proof of Digital Signature Algorithm.

Unit 4 - Network and System Security: Authentication Applications: Kerberos X-509, Directory Authentication Service, Electronic Mail Security, Pretty Good Privacy (PGP),S/Mine Security: Architecture, Authentication Header, Encapsulating Security Payloads, Combining Security Associations, Key Management, Web Security: Secure Socket Layer and Transport Layer Security, Secure Electronic Transaction (Set), System Security: Intruders, Viruses, Firewall Design Principles, Trusted Systems.

References:

1. William Stallings, "Cryptography and Network Security: Principles and Practice" Prentice Hall, New Jersey
2. Johannes A. Buchmann, "Introduction to Cryptography" Springer-Verlag
3. Atul Kahate, "Cryptography and Network Security" TMH
4. Network Security Bible: Eric Cole, Wiley dreamtech India Pvt. Ltd.
5. Practical Cryptography “Bruce Schneier” Wiley dreamtech India Pvt. Ltd.

***Note: In cases where a subject appears in both the UG Honours and UG Honours with Research programme structures, the syllabus prescribed for that subject shall be the same.**

Semester VIII (UG with Honours & UG Honours with Research*)

Software Engineering: SOS/CSE/MJ/C803

Course Objectives:

1. To understand the fundamental concepts and importance of software engineering in the development of high-quality software.
2. To familiarize students with software development life cycle models and requirement engineering.
3. To explore design principles, approaches, and coding strategies used in software development.
4. To study different software testing strategies and quality assurance techniques.
5. To understand project management aspects such as cost estimation, scheduling, risk analysis, and configuration management.
6. To introduce the use and architecture of CASE tools and environments.

Course Content:

Unit 1 - Introduction: Introduction to software engineering, Importance of software, evolving role of software, Software Characteristics, Software Components, Software Applications, Software Crisis, Software engineering problems, Software Development Life Cycle, Software Process. Software Requirement Specification: Analysis, Principles, Water Fall Model, The Incremental Model, Prototyping, Spiral Model, Role of management in software development, Role of matrices and Measurement, Problem Analysis, Requirement specification, Monitoring and Control.

Unit 2 - Software-Design: Design principles, problem partitioning, abstraction, top down and bottom up-design, Structured approach functional versus object-oriented approach, design specifications and verification, Monitoring and control, Cohesiveness, coupling, Fourth generation techniques, Functional independence, Software Architecture, Transaction and Transaction and Transform Mapping, Component level Design, Fourth Generation Techniques. Coding: Top-Down and Bottom-Up programming, structured programming, information hiding, programming style and internal documentation.

Unit 3 - Testing principles: Levels of testing, functional testing, structural testing, test plane, test case specification, reliability assessment, software testing strategies, Verification and validation, Unit testing, Integration Testing, Alpha and Beta testing, system testing and debugging.

Unit 4 - Software Project Management: The Management spectrum (The people, the product, the process, the project) Cost estimation, project scheduling, staffing, software configuration management, Structured Vs. Unstructured maintenance, quality assurance, project monitoring, risk management. Software Reliability and Quality Assurance: Reliability issues, Reliability metrics, Reliability growth modeling, Software quality, ISO 9000 Certification for software industry, SEI capability maturity model, comparison between ISO and SEI CMM.

Unit 5 - CASE (Computer Aided Software Engineering): CASE and its scope, CASE support in software life cycle, documentation, project management, internal interface, Reverse Software Engineering, Architecture of CASE environment.

References:

1. Jalote, Pankaj, "Software Engineering Ed.2"New Delhi: Narosa 2002
2. Schaum's Series, "Software Engineering" TMH
3. Ghezzi Carlo and Others "Fundamentals of Software Engineering" PHI
4. Alexis, Leon and Mathews Leon, "Fundamental of Software Engg.

***Note: In cases where a subject appears in both the UG Honours and UG Honours with Research programme structures, the syllabus prescribed for that subject shall be the same.**

Semester VIII (UG with Honours & UG Honours with Research*)

Elective VIII

Python Programming: SOS/CSE/MJ/E801

Course Objectives:

1. To understand the fundamental concepts and importance of software engineering in the development of high-quality software.
2. To familiarize students with software development life cycle models and requirement engineering.
3. To explore design principles, approaches, and coding strategies used in software development.
4. To study different software testing strategies and quality assurance techniques.
5. To understand project management aspects such as cost estimation, scheduling, risk analysis, and configuration management.
6. To introduce the use and architecture of CASE tools and environments.

Course Content

Unit-1 Introduction to Programming

Problem solving strategies; Structure of a Python program; Syntax and semantics; Executing simple programs in Python.

Unit-2 Creating Python Programs

Identifiers and keywords; Literals, numbers, and strings; Operators; Expressions; Input/output statements; Defining functions; Control structures (conditional statements, loop control statements, break, continue and pass, exit function), default arguments.

Unit-3 Built-in data structures

Mutable and immutable objects; Strings, built-in functions for string, string traversal, string operators and operations; Lists creation, traversal, slicing and splitting operations, passing list to a function; Tuples, sets, dictionaries and their operations.

Unit 4 Object Oriented Programming

Introduction to classes, objects and methods; Standard libraries

Unit 5 File and exception handling

File handling through libraries; Errors and exception handling.

References:

1. Python Programming – S. Taneja
2. Introduction to Computing and Problem-Solving using Python – E. Balaguruswamy

***Note: In cases where a subject appears in both the UG Honours and UG Honours with Research programme structures, the syllabus prescribed for that subject shall be the same.**

Semester VIII (UG with Honours & UG Honours with Research*)

Image Processing: SOS/CSE/MJ/E802

Course Objectives

1. To introduce the fundamental concepts of digital image processing and its applications.
2. To understand the basic components of an image processing system and visual perception.
3. To study techniques for image enhancement in both spatial and frequency domains.
4. To analyze image restoration models and compression techniques for efficient image storage and transmission.
5. To understand morphological operations and segmentation techniques used for object recognition and analysis.
6. To develop skills for implementing image processing algorithms for practical applications.

Course Content:

Unit 1 Introduction: Digital Image Processing, The origins of Digital Image Processing, Examples of Digital Image Processing application, Fundamental steps in Digital Image processing, Components of Image Processing system Fundamentals: Elements of Visual Perception, Light and Electromagnetic Spectrum, Image Sensing and Acquisition, Image Sampling and Quantization, Some basic Relationships between Pixels, Linear and Nonlinear Operations.

Unit 2 Image Enhancement in the spatial domain: Background, some basic gray level transformation, Introduction of Histogram processing, Enhancement using Arithmetic/Logic operations, Basics of spatial filtering, smoothing spatial filters, Sharpening spatial filters, Image Enhancement in the Frequency Domain: Introduction.

Unit 3 Image Restoration: Model of the Image Degradation/Restoration process, Noise Models, Restoration in the presence of noise only spatial filtering, Inverse filtering, Minimum Mean Square Error (Wiener) filtering, Geometric mean filter, Geometric Transformations, Image Compression: Fundamentals, Lossy Compression, Lossless Compression, Image Compression models, Error-free Compression Variable length coding, LZW coding, Bit plane coding, Run length coding, Introduction to JPEG.

Unit 4 Morphology: Dilation, Erosion, Opening and Closing, Hit-and Miss transform, Morphological Algorithms: Boundry Extraction, Region filling, Extraction of connected components, Convex Hull, Image Segmentation: Definition, characteristics of segmentation Detection of Discontinuities, Edge Linking and Boundary Detection, Thresholding, Region based segmentation. Introduction to Representation and Description, Introduction to Object Recognition.

References:

1. Digital Image Processing: Rafael C. Gonzalez and Richard E.Woods. Addision Wesley.
2. Fundamentals of Digital Image Processing. Anil K. Jain, PHI.
3. Digital Image Processing and Analysis: B. Chanda and D. Dutta Majumber, PHI.
4. Image Processing in C: Dwayne Phillips, BPB.

***Note: In cases where a subject appears in both the UG Honours and UG Honours with Research programme structures, the syllabus prescribed for that subject shall be the same.**

Semester VIII (UG with Honours & UG Honours with Research*)

SOS/CSE/MI/C804: Data Mining

Course Objectives

1. To introduce the basic concepts and necessity of data mining and data warehousing.
2. To understand the architecture and components of data warehouses and OLAP systems.
3. To explore data mining primitives and techniques for preprocessing, integration, and transformation.
4. To learn various algorithms and methods for association rule mining, classification, and prediction.
5. To study clustering techniques and their scalability for large datasets.

Course Content:

Unit 1 - Introduction to data mining, need for data warehousing and data mining, application potential, keywords and techniques. Data Warehousing and Online analytical Processing (OLAP): Aggregation operations, models for data warehousing, star schema, fact and dimension tables, conceptualization of data warehouse and multidimensional databases, Relationship between warehouse and mining.

Unit 2 - Data mining primitives: Data preprocessing, data integration, data transformation. Definition and specification of a generic data mining task. Description of Data mining query language with examples. Association analysis: Different methods for mining association rules in transaction-based data bases. Illustration of confidence and support. Multidimensional and multilevel association rules. Classification of association rules. Association rule algorithms – A priori and frequent pattern growth.

Unit 3 - Classification and Prediction: Different classification algorithms. Use of genie index, decision tree induction, Bayesian classification, neural network technique of back propagation, fuzzy set theory and genetic algorithms. Clustering: Partition based clustering, hierarchical clustering, model-based clustering for continuous and discrete data. Scalability of clustering algorithms. Parallel approaches for clustering.

Unit 4 - Web mining: Web usage mining, web content mining, web log attributes. Data mining issues in object-oriented data bases, spatial data bases and multimedia data bases and text data bases.

References:

1. J. Han, M. Kamber, “Data Mining Concepts and Techniques”, Harcourt India Pvt Ltd, 2001
2. M. Dunham, “Data Mining: introductory and Advanced Topics”, Pearson Pub, 2003
3. A.K. Pujari, “Data Mining Techniques”, Universities Press.

***Note: In cases where a subject appears in both the UG Honours and UG Honours with Research programme structures, the syllabus prescribed for that subject shall be the same.**

Semester VIII (UG with Honours & UG Honours with Research*)

SOS/CS/MJ/CP42: Compiler Design Lab
Course Content: 1. Program to check a string under a given grammar. 2. Program to check for keywords in a given string. 3. Program to check for identification for a given string. 4. Program to check for constant in a given string. 5. Program to check for relation all operators in a given string. 6. Program for lexical analyzer. 7. Implementation of stack using C 8. Implementation of shift – reduce parsing using C. 9. Implementation of three address code using quadruples.

***Note: In cases where a subject appears in both the UG Honours and UG Honours with Research programme structures, the syllabus prescribed for that subject shall be the same.**

Semester VIII (UG Honours with Research)

Research Methodology: SOS/CSE/MJ/R703

Course Objective:

To equip students with the foundational understanding of research methodologies, including the identification of research problems, formulation of research design, and application of qualitative and quantitative approaches, along with sampling and data analysis techniques.

Course Content:

Unit 1 - Meaning of Research: Function of Research, Research Characteristics, Steps involved in Research, Significance of Research, Types of Research, Criteria of Good Research, Research in Pure and Applied Sciences, Areas of Science, Philosophy of Science, Interdisciplinary Research, Review of Literature.

Unit 2 - Identification of Research Problem: Selecting the Research Problem, Defining the Problem, Goals and Criteria for Identifying problems for research, Techniques Involved in Defining the Problem, Source of Problems, Personal Consideration.

Unit 3 - Research Design: Concept and Importance in Research, Need for Research Design,

Formulation of Research Design, Exploratory Research Design, Descriptive Research Designs, Experimental Design, Basic principles of experimental designs, Computer and Internet in designs.

Unit 4 - Qualitative and Quantitative Research: Qualitative research, Quantitative research, Concept of measurement, causality, generalization, replication, Merging the two approaches.

Unit 5 - Sampling and Data Analysis: Sampling Frame, Sample Size, Characteristics of a Good Sample, Random Sampling, Systematic Sampling, Practical Considerations in Sampling and Sample Size, Data Preparation, Univariate and Bivariate analyses.

Reference Books:

Research Methodology: Methods & Techniques, C.R.Kothari, New Age International Publishers.

Research Design: Qualitative, Quantitative, and Mixed Methods Approaches, John W. Creswell, SAGE Publications Inc.

Research Methodology, Mukul Gupta & Deepa Gupta, PHI Learning Private Ltd, New Delhi.

Research Methodology, Uma Sekaran & Roger Bougie, NMIMS Global Access 5. The Scientific Endeavor, Jeffrey A. Lee, Pearson India

***Note: In cases where a subject appears in both the UG Honours and UG Honours with Research programme structures, the syllabus prescribed for that subject shall be the same.**