B.Tech. – III Year I Semester

S.No.	Category	Title	L	T	P	Credits
1	Professional Core	Cloud Computing	3	0	0	3
2	Professional Core	Introduction to Cyber Security	3	0	0	3
3	Professional Core	Automata Theory & Compiler Design	3	0	0	3
4	Professional Elective-I	1. Software Engineering 2. Wireless Sensor Networks 3. Artificial Intelligence 4. Internet of Things 5. 12-week MOOC Swayam/NPTEL course recommended by the BoS	3	0	0	3
5	Open Elective- I		3	0	0	3
6	Professional Core	Cloud Computing Lab	0	0	3	1.5
7	Professional Core	Cyber Security Lab	0	0	3	1.5
8	Skill Enhancement course	Full Stack Development-2	0	1	2	2
9	Engineering Science	Ui Design-Flutter Lab	0	0	2	1
10	Evaluation of Community Service Internship		-	-	-	2
		Total	**15**	**1**	**10**	**23**

Minor Courses & Honor Course

S.No.	Category	L	T	P	Credits
1	Minor Course (Student may select from the specialized minors pool)	3	0	0	3
2	Minor Course through SWAYAM/NPTEL (minimum 12week, 3 credit course)	3	0	0	3
3	Honors Course (Student may select from the honors pool)	3	0	0	3

B.Tech.–III Year II Semester

S.No.	Category	Title	L	T	P	Credits
1	Professional Core	Cyber Crimes & Digital Forensics	3	0	0	3
2	Professional Core	Cryptography & Network Security	3	0	0	3
3	Professional Core	Machine Learning	3	0	0	3
4	Professional Elective-II	1. Software Testing Methodologies 2. DevOps 3. Microprocessors & Microcontrollers 4. Applied Cryptography 5. 12-week MOOC Swayam/NPTEL course recommended by the BoS	3	0	0	3
5	Professional Elective-III	1. Software Project Management 2. Mobile Adhoc Networks 3. Natural Language Processing 4. Security Assessment and Risk Analysis 5. 12-week MOOC Swayam/NPTEL course recommended by the BoS	3	0	0	3
6	Open Elective – III		3	0	0	3
7	Professional Core	Cryptography & Network Security Lab	0	0	3	1.5
8	Professional Core	Cyber Crimes & Digital Forensics Lab	0	0	3	1.5
9	Skill Enhancement course	Soft skills OR IELTS	0	1	2	2
10	Audit Course	Technical Paper Writing & IPR	2	0	0	0
		Total	**20**	**1**	**08**	**23**
Mandatory Industry Internship of 08 weeks duration during summer vacation						

III Year I Semester	CLOUD COMPUTING	L	T	P	C
		3	0	0	3

Course Objectives:

- To explain the evolving utility computing model called cloud computing.
- To introduce the various levels of services offered by cloud.
- To discuss the fundamentals of cloud enabling technologies such as distributed computing, service-oriented architecture and virtualization.
- To emphasize the security and other challenges in cloud computing.
- To introduce the advanced concepts such as containers, serverless computing and cloud-centric Internet of Things.

Course Outcomes:

Upon the completion of the course, the students will be able to:

CO 1: Explain the fundamental concepts of cloud computing, cloud service models, deployment models, characteristics, benefits, and major cloud service providers.

CO 2: Explain the fundamental concepts of cloud computing, cloud service models, deployment models, characteristics, benefits, and major cloud service providers.

CO 3: Analyze virtualization techniques and containerization platforms to evaluate their role, advantages, and limitations in cloud environments.

CO 4: Assess cloud computing challenges such as security, interoperability, scalability, energy efficiency, and apply appropriate security models and standards for cloud deployment scenarios.

CO 5: Design solutions using advanced cloud computing concepts like serverless computing, IoT integration, edge/fog computing, DevOps, and quantum cloud computing for real-world applications.

UNIT -I:

Introduction to Cloud Computing Fundamentals, Cloud computing at a glance, defining a cloud, cloud computing reference model, types of services (IaaS, PaaS, SaaS), cloud deployment models (public, private, hybrid), utility computing, cloud computing characteristics and benefits, cloud service providers (Amazon Web Services, Microsoft Azure, Google AppEngine).

UNIT-II

Cloud Enabling Technologies, Ubiquitous Internet, parallel and distributed computing, elements of parallel computing, hardware architectures for parallel computing (SISD, SIMD, MISD, MIMD), elements of distributed computing, Inter-process communication, technologies for distributed computing, remote procedure calls (RPC), service-oriented architecture (SOA), Web services, virtualization.

UNIT-III

Virtualization and Containers, Characteristics of virtualized environments, taxonomy of virtualization techniques, virtualization and cloud Computing, pros and cons of virtualization, technology examples (XEN, VMware), building blocks of containers, container platforms (LXC, Docker), container orchestration, Docker Swarm and Kubernetes, public cloud VM (e.g. Amazon EC2) and container (e.g. Amazon Elastic Container Service) offerings.

UNIT-IV:

Cloud computing challenges, Economics of the cloud, cloud interoperability and standards, scalability and fault tolerance, energy efficiency in clouds, federated clouds, cloud computing security, fundamentals of computer security, cloud security architecture, cloud shared responsibility model, security in cloud deployment models.

UNIT -V

Advanced concepts in cloud computing, Serverless computing, Function-as-a-Service, serverless computing architecture, public cloud (e.g. AWS Lambda) and open-source (e.g. OpenFaaS) serverless platforms, Internet of Things (IoT), applications, cloud-centric IoT and layers, edge and fog computing, DevOps, infrastructure-as-code, quantum cloud computing.

Mapping of COs to POs:

POs/ COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PS O1	PS O2
CO1	3	2	2	1	3	1	–	–	–	1	–	2	3
CO2	3	2	2	1	3	1	–	–	–	1	–	2	3
CO3	3	3	3	2	3	–	–	–	1	1	–	3	3
CO4	3	3	3	3	3	2	2	2	1	1	–	2	3
CO5	3	3	3	3	3	1	–	–	1	2	1	3	3

Text Books:

1. Mastering Cloud Computing, 2nd edition, Rajkumar Buyya, Christian Vecchiola, ThamaraiSelvi, ShivanandaPoojara, Satish N. Srirama, McGraw Hill, 2024.
2. Distributed and Cloud Computing, Kai Hwang, Geoffery C. Fox, Jack J. Dongarra, Elsevier, 2012.
3. Cloud Computing: Concepts, Technology, Security & Architecture, Thomas Erl, Eric Barceló Monroy, 2nd Edition - Pearson , 2024.

Reference Books:

1. Cloud Computing, Theory and Practice, Dan C Marinescu, 2nd edition, MK Elsevier, 2018.
2. Essentials of cloud Computing, K. Chandrasekhran, CRC press, 2014.
3. Online documentation and tutorials from cloud service providers (e.g., AWS, Azure, GCP).
4. Cloud Computing & Big Data: From the Basics to Practical Use Cases, M Sudheep Elayidom, Sarith Divakar M, Lija Mohan, Tanmay Kumar Pandey, 1st Edition, Cengage, 2024.

III Year **I Semester**	**INTRODUCTION TO CYBER SECURITY**	**L**	**T**	**P**	**C**
		3	**0**	**0**	**3**

Course Objectives:

- Understand the threats in networks and security concepts.
- Apply authentication applications in different networks.
- Understand security services for email.
- Awareness of firewall and it applications.

Course Outcomes:

Upon the completion of the course, the students will be able to:

CO 1: Explain the fundamentals of information security, best practices, safe internet usage, and securing computer networks.

CO 2: Understand the ethical, legal, and social issues related to cyber security, including cyber laws, privacy, intellectual property, and cybercrimes.

CO 3: Apply penetration testing methodologies to assess web application security, identify vulnerabilities, and define web testing scopes.

CO 4: Analyze web application security issues such as XSS, SQL Injection, CSRF, Session Hijacking, and perform forensic investigations to ensure network assurance.

CO 5: Apply risk management strategies to evaluate assets, quantify risks, develop security policies, and ensure business continuity in cyber incident response.

UNIT-I

Introduction to Information Security Fundamentals and Best Practices: Protecting Your Computer and its Contents, Securing Computer Networks--Basics of Networking, Compromised Computers, Secure Communications and Information Security Best Practices, Privacy Guidelines, Safe Internet Usage.

UNIT-II

Ethics in Cyber Security & Cyber Law: Privacy, Intellectual Property, Professional Ethics, Freedom of Speech, Fair User and Ethical Hacking, Trademarks, Internet Fraud, Electronic Evidence, Cybercrimes.

UNIT-III

Penetration Testing: Overview of the web from a penetration testers perspective, Exploring the various servers and clients, Discussion of the various web architectures, Discussion of the different types of vulnerabilities, defining a web application test scope and process, Defining types of penetration testing.

UNIT-IV

Web Application Security: Common Issues in Web Apps, what is XSS, SQL injection, CSRF, Password Vulnerabilities, SSL, CAPTCHA, Session Hijacking, Local and Remote File Inclusion, Audit Trails, Web Server Issues. **Forensics & Network Assurance:** Forensic Technologies, Digital Evidence Collection, Evidentiary Reporting, Layered Defense, Surveillance and Reconnaissance, Outsider Thread Protection

UNIT-V

Information Risk Management: Asset Evaluation and Business Impact Analysis, Risk Identification, Risk Quantification, Risk Response Development and Control, Security Policy, Compliance, and Business Continuity. Forensic investigation using Access Data FTK, En-Case, **Cyber Incident Analysis and Response:** Incident Preparation, Incident Detection and Analysis. Containment, Eradication, and Recovery. Proactive and Post-Incident Cyber Services, CIA triangle

Mapping of COs to POs:

POs/ COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	2	2	3	2	1	–	–	2	–	2	3
CO2	2	2	–	1	2	3	2	3	1	2	2	–	2
CO3	3	3	3	3	3	–	–	–	1	2	–	3	3
CO4	3	3	3	3	3	1	–	–	–	2	2	3	3
CO5	3	3	3	3	3	3	2	2	1	2	2	3	3

Text Books:

1. Cyber Security & Digital Forensics by Anas Zakir, Clever Fox Publishing, Publication Date- 2022
2. “Beginners Guide To Ethical Hacking and Cyber Security “, by Abhinav Ojha, Khanna Publishers, First Edition, Publication Date-2023.
3. Introduction to Information Security and Cyber Laws, Surya Prakash Tripati, Ritendra Goel, Praveen Kumar Shukla, Dreamtech Press,

Reference Books

1. The Official CHFI Study Guide for Computer Hacking Forensic Investigator by Dave Kleiman
2. CISSP Study Guide, 6th Edition by James M. Stewart.
3. Cyber Forensics , Dejey, Murugan, First Edition, Oxford, 2018

III Year I Semester	AUTOMATA THEORY & COMPILER DESIGN	L	T	P	C
		3	0	0	3

Course Objectives:

- To introduce the. Fundamental concepts of formal languages, grammars and automata theory.
- To understand deterministic and non-deterministic machines and the differences between decidability and undecidability.
- Introduce the major concepts of language translation and compiler design and impart the knowledge of practical skills necessary for constructing a compiler.
- Topics include phases of compiler, parsing, syntax directed translation, type checking use of symbol tables, intermediate code generation.

Course Outcomes:

Upon the completion of the course, the students will be able to:

CO 1: Explain formal language theory, regular expressions, finite automata (DFA, NFA), and context-free grammars with their applications in lexical analysis and parsing.

CO 2: Apply bottom-up parsing techniques such as LR and LALR parsing, syntax-directed translation, and use YACC specifications for parsing and intermediate code generation.

CO 3: Analyze the context-sensitive features of programming languages, including type checking, type conversions, and the Chomsky hierarchy, to validate language constructs.

CO 4: Evaluate run-time storage organization, storage allocation strategies, and code optimization techniques to improve program performance and memory utilization.

CO 5: Develop efficient machine-level code using code generation algorithms, register allocation strategies, and DAG-based block representations.

UNIT -I

Formal Language and Regular Expressions: Languages, Definition Languages regular expressions, Finite Automata-DFA, NFA. Conversion of regular expression to NFA, NFA to DFA. Applications of Finite Automata to lexical analysis, lex tools. **Context Free grammars and parsing:** Context free grammars, derivation, parse trees, ambiguity LL(K) grammars and LL.(1) parsing

UNIT-II

Bottom up parsing handle pruning LR Grammar Parsing, LALR parsing, parsing ambiguous grammars, YACC programming specification.

Semantics : Syntax directed translation, S-attributed and L-attributed grammars, Intermediate code - abstract syntax tree, translation of simple statements and control flow statements.

UNIT-III

Context Sensitive features - Chomsky hierarchy of languages and recognizers. Type checking, type conversions, equivalence of type expressions, overloading of functions and operations.

UNIT-IV

Run time storage : Storage organization, storage allocation strategies scope access to now local names. parameters, language facilities for dynamics storage allocation.

Code optimization : Principal sources of optimization, optimization of basic blocks, peephole optimization, flow graphs, Data flow analysis of flow graphs.

UNIT-V

Code generation : Machine dependent code generation, object code forms, generic code generation algorithm, Register allocation and assignment. Using DAG representation of Block.

Mapping of COs to POs:

POs/ COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PS O1	PS O2
CO1	3	3	2	2	3	–	–	–	–	2	–	3	2
CO2	3	3	3	3	3	–	–	–	–	2	–	3	2
CO3	3	3	2	2	2	–	–	–	–	1	–	2	2
CO4	3	3	3	3	3	–	–	–	–	1	–	3	2
C05	3	3	3	3	3	–	–	–	–	2	–	3	2

TEXT BOOKS:

1. Introduction to Theory of computation. Sipser, 2nd Edition, Thomson.

2. Compilers Principles, Techniques and Tools Aho, Ullman, Ravisethi, Pearson Education.

3. Modern Compiler Design, Dick Grune, Kees van Reeuwijk, Henri E. Bal, Ceriel J.H. Jacobs, Springer, 2016

REFERENCES:

1. Modern Compiler Construction in C, Andrew W.Appel Cambridge University Press.

2.CompilerConstruction,LOUDEN,Thomson.

3. THEORY OF COMPUTATION Paperback – 1 January 2024 by Shripriti Publications

III Year I Semester	SECURE CODING PRACTICES	L	T	P	C
		3	0	0	3

Course Objectives:

1. To understanding of the various security attacks and knowledge to recognize and remove common coding errors that lead to vulnerabilities.
2. It gives an outline of the techniques for developing a secure application.

Course Outcomes:

Upon the completion of the course, the students will be able to:

CO 1: Explain the need for secure software systems, security considerations across the software development life cycle (S-SDLC), and best practices for writing secure code using SD3 principles.

CO 2: Apply secure coding techniques to protect against common software vulnerabilities such as DoS attacks, buffer overruns, code injection, and memory management issues in Java and C.

CO 3: Analyze threat modeling techniques, risk assessment processes like DREAD, and security strategies including defense in depth and the principle of least privilege to mitigate software security threats.

CO 4: Evaluate web-specific security issues such as SQL injection, race conditions, XSS attacks, and implement secure input validation and secure testing practices for different application types.

CO 5: Design secure software systems using security-aware requirements engineering (SQUARE process), misuse/abuse cases, and software security best practices for architecture and design.

UNIT- I

INTRODUCTION: Need for secure systems: Proactive Security development process, Secure Software Development Cycle (S-SDLC), Security issues while writing SRS, Design phase security, Development Phase, Test Phase, Maintenance Phase, Writing Secure Code – Best Practices SD3 (Secure by design, default and deployment), Security principles and Secure Product Development Timeline

UNIT -II

SECURE CODING TECHNIQUES: Protection against DoS attacks, Application Failure Attacks, CPU Starvation Attacks, Insecure Coding Practices in Java Technology. ARP Spoofing and its countermeasures. Buffer Overrun- Stack overrun, Heap Overrun, Array Indexing Errors, Format String Bugs. Security Issues in C Language: String Handling, Avoiding Integer Overflows and Underflows and Type Conversion Issues- Memory Management Issues, Code Injection Attacks, Canary based countermeasures using StackGuard and Propolice. Socket Security, Avoiding Server Hijacking, Securing RPC, ActiveX and DCOM

UNIT–III

Threat modelling process and its benefits: Identifying the Threats by Using Attack Trees and rating threats using DREAD, Risk Mitigation Techniques and Security Best Practices. Security techniques, authentication, authorization. Defense in Depth and Principle of Least Privilege.

UNIT-IV

AND WEB SPECIFIC INPUT ISSUES: SQL Injection Techniques and Remedies, Race conditions, Time of Check Versus Time of Use and its protection mechanisms. Validating Input and Inter process Communication, Securing Signal Handlers and File Operations. XSS scripting attack and its types – Persistent and Non persistent attack XSS Countermeasures and Bypassing the XSS Filters. Testing Secure Applications: Security code overview, secure software installation. The Role of the Security Tester, Building the Security Test Plan. Testing HTTP-Based Applications, Testing File-Based Applications, Testing Clients with Rogue Servers

UNIT -V

SOFTWARE SECURITY ENGINEERING: Requirements engineering for secure software: Misuse and abuse cases- SQUARE process model- Software security practices and knowledge for architecture and design.

Mapping of COs to POs:

POs/ COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PS O1	PS O2
CO1	3	2	2	2	2	2	2	–	–	2	2	3	2
CO2	3	3	3	2	3	–	–	–	–	1	–	3	2
CO3	3	3	2	3	2	2	2	2	–	2	2	3	2
CO4	3	3	3	3	3	–	–	–	–	1	–	3	2
C05	3	3	3	3	3	2	2	2	1	2	2	3	2

Text Books:

1. Writing Secure Code, Michael Howard and David LeBlanc, Microsoft Press, 2nd Edition, 2004.
2. Threat Modeling, Frank Swiderski and Window Snyder, Microsoft Professional, 1st Edition, 2004.
3. Secure Coding: Fundamental Principles and Best Practices, Mr. Shaik Abdul Subhahan , Mr. Nazeer Shaik, Dr. C. Krishna Priya, Alize Software Services LLP , 2025

Reference Books:

1. Robert C.Seacord, “ *Secure Coding in C and C++*”, Pearson Education, 2nd edition, 2013.
2. Julia H. Allen, Sean J. Barnum, Robert J. Ellison, Gary McGraw, Nancy R. Mead, “ *Software Security Engineering : A guide for Project Managers*”, Addison-Wesley Professional, 2008.
3. Buffer Overflow Attacks: Detect, Exploit, Prevent by Jason Deckar, Syngress, 1st Edition, 2005.
4. Secure Coding Practices: Fortifying Applications Against Cyber Threats, Black Hat Kathy, Telephasic Workshop , 2024

III Year I Semester	SOFTWARE ENGINEERING (Professional Elective-I)	L	T	P	C
		3	0	0	3

Course Objectives:

The objectives of this course are to introduce

- Software life cycle models
- Software requirements and SRS document.
- How to plan for a project.
- The quality control and how to ensure good quality software.
- Testing **Course** methods of software, use of CASE tools
- Implementation issues, validation and verification procedures.

Course Outcomes:

Upon the completion of the course, the students will be able to:

CO 1: Explain the evolution of software engineering, various software development life cycle models, and modern software development practices including Agile and Spiral models.

CO 2: Apply software project management techniques, project size estimation methods, risk management strategies, and software requirement specification techniques to manage software projects effectively.

CO 3: Analyze software design principles including cohesion and coupling, function-oriented design methodologies, structured analysis using DFDs, and user interface design techniques.

CO 4: Evaluate coding practices, software testing methods, software reliability measures, and quality management standards such as ISO 9000 and SEI CMM for software quality assurance.

CO 5: Develop software solutions using Computer-Aided Software Engineering (CASE) tools, implement effective software maintenance strategies, and design reusable software components within an organizational reuse program.

UNIT-I

INTRODUCTION: Evolution, Software development projects, Exploratory style of software developments, Emergence of software engineering, Notable changes in software development practices, Computer system engineering. **SOFTWARE LIFE CYCLE MODELS:** Basic concepts, Waterfall model and its extensions, Rapid application development, Agile development model and Spiral model.

UNIT -II:

SOFTWARE PROJECT MANAGEMENT: Software project management complexities, Responsibilities of a software project manager, Metrics for project size estimation, Project estimation techniques, Empirical Estimation techniques, COCOMO, Halstead’s software science, and risk management. **REQUIREMENTS ANALYSIS AND SPECIFICATION:** Requirements gathering and analysis, Software Requirements Specification (SRS), Formal system specification, Axiomatic specification, Algebraic specification, Executable specification and 4GL.

UNIT III:

SOFTWARE DESIGN: Overview of the design process, How to characterise a good software design? Layered arrangement of modules, Cohesion and Coupling. approaches to software design. **FUNCTION-ORIENTED SOFTWARE DESIGN:** Overview of SA/SD methodology, Structured analysis, Developing the DFD model of a system, Structured design, Detailed design, and Design Review. **USER INTERFACE DESIGN:** Characteristics of a good user interface, Basic concepts, Types of user interfaces, Fundamentals of component-based GUI development, and user interface design methodology.

UNIT IV:

CODING AND TESTING: Coding, Code review, Software documentation, Testing, Black-box testing, White-Box testing, Debugging, Program analysis tools, Integration testing, Testing object-oriented programs, Smoke testing, and Some general issues associated with testing. **SOFTWARE RELIABILITY AND QUALITY MANAGEMENT:** Software reliability. Statistical testing, Software quality, Software quality management system, ISO 9000.SEI Capability maturity model. Few other important quality standards, and Six Sigma.

UNIT V:

COMPUTER-AIDED SOFTWARE ENGINEERING (CASE): CASE and its scope, CASE environment, CASE support in the software life cycle, Other characteristics of CASE tools, Towards second generation CASE Tool, and Architecture of a CASE Environment. **SOFTWARE MAINTENANCE:** Characteristics of software maintenance, Software reverse engineering, Software maintenance process models and Estimation of maintenance cost. **SOFTWARE REUSE:** What can be reused? Why almost no reuse so far? Basic issues in any reuse program, A reuse approach, and Reuse at organisation level.

Mapping of COs to POs:

POs/ COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PS O1	PS O2
CO1	3	2	2	1	2	2	1	–	1	2	2	2	1
CO2	3	3	3	2	3	2	–	–	1	2	3	3	1
CO3	3	3	3	2	2	–	–	–	–	2	–	2	1
CO4	3	3	3	3	3	2	–	–	–	2	2	3	2
C05	3	3	3	3	3	–	–	–	1	2	3	3	2

Text Books

1. Fundamentals of Software Engineering, Raji b Mall, Fifth Edition, PHI.
2. Software Engineering for Automotive Systems: Principles and Applications, B. Vinoth Kumar, P. Sivakumar, R. S. Sandhya Devi, CRC Press; 1st edition, 2022

Reference Books

1. Software Engineering Apractitioner’s Approach, Roger S. Pressman, Ninth Edition, McGraw Hill International Edition.
2. Software Engineering, Ian Sommerville, Tenth Edition, Pearson Education.
3. Software Engineering, Principles and Practices, Deepak Jain, Oxford University Press.
4. New Software Engineering Paradigm Based on Complexity Science: An Introduction to NSE, Jay Xiong, Springer, 2014.

e-Resources:

1) https://nptel.ac.in/courses/106/105/106105182/
2) https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01260589506387148827_shared/overview
3) https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013382690411003904735_shared/overview

III Year I Semester	WIRELESS SENSOR NETWORKS (Professional Elective-I)	L	T	P	C
		3	0	0	3

Course Objectives:

- To acquire the knowledge about various architectures and applications of Sensor Networks
- To understand issues, challenges, and emerging technologies for wireless sensor networks
- To learn about various routing protocols and MAC Protocols
- To understand various data gathering and data dissemination methods
- To study about design principles, node architectures, hardware, and software required for implementation of wireless sensor networks.

Course Outcomes:

Upon the completion of the course, the students will be able to:

CO 1: Explain the fundamental concepts, challenges, advantages, types, and applications of Wireless Sensor Networks.

CO 2: Describe the role of Mobile Ad-hoc Networks (MANETs), enabling technologies, and the key issues and challenges associated with the design and deployment of Wireless Sensor Networks.

CO 3: Apply routing and Medium Access Control (MAC) protocols, including S-MAC, B-MAC, IEEE 802.15.4, and ZigBee, for efficient communication in Wireless Sensor Networks.

CO 4: Analyze data dissemination, data fusion, real-time traffic, and security protocols to improve the performance and quality of Wireless Sensor Networks.

CO 5: Design Wireless Sensor Network architectures, including gateway connectivity, hardware components, operating systems, and programming environments like TinyOS and nesC.

UNIT-I

Introduction to Sensor Networks, unique constraints and challenges, Advantage of Sensor Networks, Applications of Sensor Networks, Types of wireless sensor networks

UNIT-II

Mobile Ad-hoc Networks (MANETs) and Wireless Sensor Networks, Enabling technologies for Wireless Sensor Networks. Issues and challenges in wireless sensor networks

UNIT-III

Routing protocols, MAC protocols: Classification of MAC Protocols, S-MAC Protocol, B-MAC protocol, IEEE 802.15.4 standard and ZigBee

UNIT-IV

Dissemination protocol for large sensor network. Data dissemination, data gathering, and data fusion; Quality of a sensor network; Real-time traffic support and security protocols.

UNIT-V

Design Principles for WSNs, Gateway Concepts, Need for gateway, WSN to Internet Communication, and Internet to WSN Communication. Single-node architecture, Hardware components & design constraints, Operating systems and execution environments, introduction to TinyOS and nesC.

Mapping of COs to POs:

POs/ COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PS O1	PS O2
CO1	3	2	2	2	2	–	–	–	–	1	–	2	2
CO2	3	2	2	2	2	2	2	–	–	1	–	2	3
CO3	3	3	3	3	3	–	–	–	–	1	–	3	3
CO4	3	3	3	3	3	2	2	2	–	2	2	3	3
C05	3	3	3	3	3	–	–	–	1	2	2	3	3

Text Books:

1. Ad-Hoc Wireless Sensor Networks - C. Siva Ram Murthy, B.S. Manoj, Pearson
2. Principles of Wireless Networks – Kaveh Pahlavan and P. Krishna Murthy, 2002, PE.
3. Ad-Hoc and Wireless Sensor network, SHASHIKANT V. ATHAWALE, Pearson Education, 1st edition, 2022

Reference Books:

1. Wireless Digital Communications – Kamilo Feher, 1999, PHI.
2. Wireless Communications - Andrea Goldsmith, 2005, Cambridge University Press.
3. Mobile Cellular Communication – Gottapu Sasibhushana Rao, Pearson Education, 2012.
4. Wireless Communication and Networking – William Stallings, 2003, PHI.
5. Fundamentals of Wireless Sensor Networks: Theory and Practice, Christian Poellabauer Waltenegus Dargie, Wiley ,2014.

III Year I Semester	ARTIFICIAL INTELLIGENCE (Professional Elective-I)	L	T	P	C
		3	0	0	3

Course Objective:

- The student should be made to study the concepts of Artificial Intelligence.
- The student should be made to learn the methods of solving problems using Artificial Intelligence.
- To understand the applications of AI, namely game playing, theorem proving, and machine learning.
- To learn different knowledge representation techniques

Course Outcomes:

Upon the completion of the course, the students will be able to:

CO 1: Explain the fundamental concepts of Artificial Intelligence, problem-solving using state space search, intelligent agents, and various agent architectures.

CO 2: Apply uninformed and heuristic search strategies, including breadth-first, depth-first, A*, AO*, and local search algorithms to solve AI-related problems efficiently.

CO 3: Analyze constraint satisfaction problems, adversarial search, and game playing strategies using techniques like minimax and alpha-beta pruning to make optimal decisions.

CO 4: Describe knowledge representation techniques using logic, rules, and reasoning methods, including predicate logic, forward and backward chaining, and logic programming.

CO 5: Evaluate probabilistic reasoning models, planning systems, and expert systems for effective decision-making in uncertain environments and complex AI applications.

UNIT-I

Introduction, Overview of Artificial intelligence: Problems of AI, AI technique, Tic - Tac - Toe problem. Intelligent Agents, Agents & environment, nature of environment, structure of agents, goal-based agents, utility-based agents, learning agents. **Problem Solving, Problem Space & search:** Defining the problem as state space search, production system, problem characteristics, issues in the design of search programs.

UNIT-II

Search techniques: Problem solving agents, searching for solutions; uniform search strategies: breadth first search, depth first search, depth limited search, bidirectional search, comparing uniform search strategies. Heuristic search strategies Greedy best-first search, A* search, AO* search, memory bounded heuristic search: local search algorithms & optimization problems: Hill climbing search, simulated annealing search, local beam search

UNIT-III

Constraint satisfaction problems: Local search for constraint satisfaction problems. Adversarial search, Games, optimal decisions & strategies in games, the minimax search procedure, alpha-beta pruning, additional refinements, iterative deepening.

UNIT – IV

Knowledge & reasoning: Knowledge representation issues, representation & mapping, approaches to knowledge representation. Using predicate logic, representing simple fact in logic, representing instant & ISA relationship, computable functions & predicates, resolution, natural deduction. Representing knowledge using rules, Procedural verses declarative knowledge, logic programming, forward verses backward reasoning, matching, control knowledge.

UNIT – V

Probabilistic reasoning: Representing knowledge in an uncertain domain, the semantics of Bayesian networks, Dempster-Shafer theory, Planning Overview, components of a planning system, Goal stack planning, Hierarchical planning, other planning techniques. **Expert Systems:** Representing and using domain knowledge, expert system shells, and knowledge acquisition.

Mapping of COs to POs:

POs/ COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PS O1	PS O2
CO1	3	2	2	2	2	–	–	–	–	1	–	2	2
CO2	3	3	3	3	3	–	–	–	–	1	–	3	3
CO3	3	3	3	3	3	–	–	–	–	2	–	3	3
CO4	3	3	2	2	2	–	–	–	–	2	–	2	2
CO5	3	3	3	3	3	–	–	–	–	2	2	3	3

Text Books:

1. Stuart Russell and Peter Norvig, Artificial Intelligence: A Modern Approach
2. Artificial Intelligence, Russel, Pearson.
3. AI for Everyone: A Beginner's Handbook for Artificial Intelligence, Saptarsi Goswami, Amit Kumar Das, Amlan Chakrabarti, Pearson, 2024

Reference Books:

1. Artificial Intelligence, Ritch & Knight, TMH
2. Introduction to Artificial Intelligence & Expert Systems, Patterson, PHI
3. Logic & Prolog Programming, Saroj Kaushik, New Age International
4. Expert Systems, Giarranto, VIKAS.
5. Artificial Intelligence For Dummies, John Paul Mueller, Luca Massaron,, Stephanie Diamond, 3rd Edition, Wiley Publications, 2025.

III Year I Semester	INTERNET OF THINGS (Professional Elective-I)	L	T	P	C
		3	0	0	3

Course Objectives:

From the course the student will learn

- the application areas of IOT
- the revolution of Internet in Mobile Devices, Cloud & Sensor Networks
- building blocks of Internet of Things and characteristics

Course Outcomes:

Upon the completion of the course, the students will be able to:

CO 1: Explain the evolution, enabling technologies, and networking components of IoT along with addressing strategies and their relationship to Wireless Sensor Networks and Cyber Physical Systems.

CO 2: Apply IoT sensing, actuation principles, and processing topologies to design IoT-enabled systems with appropriate device selection and processing models.

CO 3: Describe various IoT connectivity and communication technologies, including protocols and standards used for device discovery, data exchange, and device management.

CO 4: Analyze IoT interoperability challenges and evaluate the role of fog computing architecture and its applications in enhancing IoT systems.

CO 5: Evaluate emerging IoT paradigms, associated challenges, and develop solutions through case studies in domains like agriculture and vehicular networks.

UNIT I:

Predecessors of IoT: Introduction, Wireless Sensor Networks, Machine-to-Machine Communications, Cyber Physical Systems, Emergence of IoT: Introduction, Evolution of IoT, Enabling IoT and the Complex Interdependence of Technologies, IoT Networking Components, Addressing Strategies in IoT

UNIT II:

IoT Sensing and Actuation: Introduction, Sensors, Sensor Characteristics, Sensorial Deviations, Sensing Types, Sensing Considerations, Actuators, Actuator Types, Actuator Characteristics, IoT Processing Topologies and Types: Data Format, Importance of Processing in IoT, Processing Topologies, IoT Device Design and Selection Considerations, Processing Offloading.

UNIT III:

IoT Connectivity Technologies: Introduction, IEEE 802.15.4, Zigbee, Thread, ISA100.11A, Wireless HART, RFID, NFC, DASH7, Z-Wave, Weightless, Sigfox, LoRa, NB-IT, Wi-Fi, Bluetooth, IoT Communication Technologies: Introduction, Infrastructure Protocols, Discovery Protocols, Data Protocols, Identification Protocols, Device Management, Semantic Protocols.

UNIT IV:

IoT Interoperability: Introduction, Standards, Frameworks, Fog Computing and Its Applications: Introduction, View of Fog Computing Architecture, Fog Computing in IoT, Selected Applications of Fog Computing

UNIT V:

Paradigms, Challenges, and the Future: Introduction, Evolution of New IoT Paradigms, Challenges Associated with IoT, Emerging Pillars of IoT, IoT Case Studies: Agricultural IoT, Vehicular IoT

Mapping of COs to POs:

POs/ COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PS O1	PS O2
CO1	3	2	2	2	3	–	2	–	–	1	–	2	2
CO2	3	3	3	3	3	–	–	–	–	1	–	3	3
CO3	3	3	2	2	3	–	–	–	–	2	–	3	3
CO4	3	3	3	3	3	2	2	–	–	2	2	3	3
CO5	3	3	3	3	3	2	2	–	–	2	2	3	3

Text Books:

1. Introduction to IoT, Sudip Misra, Anandarup Mukhaerjee, Arjit Roy, Cambridge University Press, 2021
2. Internet of Things: Architecture, Design Principles and Applications, Rajkamal, McGraw Hill Higher Education.
3. Internet of Things, Shriram K Vasudevan, Abhishek S Nagarajan, RMD Sundaram, 2ed, Wiley Publications, 2020

Reference Books:

1. Fog and Edge Computing: Principles and Paradigms, Rajkumar Buyya (Editor), Satish narayana Srirama (Editor) , ISBN: 978-1-119-52498-4, January 2019
2. Getting Started with the Internet of Things, CunoPfister , Oreilly.
3. INTERNET OF THINGS - A HANDS-ON APPROACH, Arsheep Bahga, Vijay Madisetti, Orient Blackswan Private Limited - New Delhi; First Edition , 2015

III Year I Semester	CLOUD COMPUTING LAB	L	T	P	C
		0	0	3	1.5

Course Objectives:

- To introduce the various levels of services offered by cloud.
- To give practical knowledge about working with virtualization and containers.
- To introduce the advanced concepts such as serverless computing and cloud simulation.

Course Outcomes:

Upon the completion of the course, the students will be able to:

CO 1: Set up and configure virtualization environments using VirtualBox, VMware, or OpenStack and deploy multiple operating systems on a host system.

CO 2: Deploy cloud-based services on platforms such as Amazon EC2, Google App Engine, Docker, and OpenStack, including web server configuration and security management.

CO 3: Explain the working of inter-process communication (IPC), message passing, and publish/subscribe systems in distributed environments.

CO 4: Demonstrate file sharing and data exchange between virtual machines and Docker containers in a cloud environment.

CO 5: Install, configure, and execute applications on Hadoop and OpenFaaS platforms and simulate cloud scenarios using CloudSim with custom scheduling algorithms.

List of Experiments:

1. Lab on web services
2. Lab on IPC, messagaging, publish/subscribe
3. Install VirtualBox/VMware Workstation with different flavours of Linux or windows OS on top of windows8 or above.
4. Install a C compiler in the virtual machine created using VirtualBox and execute Simple Programs.
5. Create an Amazon EC2 instance and set up a web-server on the instance and associate an IP address with the instance. In the process, create a security group allowing access to port 80 on the instance.

   OR
6. Do the same with OpenStack
7. Install Google App Engine. Create a hello world app and other simple web applications using python/java.
8. Start a Docker container and set up a web-server (e.g. apache2 or Python based Flask micro web framework) on the instance. Map the host directory as a data volume for the container.
9. Find a procedure to transfer the files from one virtual machine to another virtual machine. Similarly, from one container to another container.
10. Find a procedure to launch virtual machine using trystack (Online Openstack Demo Version)
11. Install Hadoop single node cluster and run simple applications like word count.
12. Utilize OpenFaaS – Serverless computing framework and demonstrate basic event driven function invocation.
13. Simulate a cloud scenario using CloudSim and run a scheduling algorithm that is not present in CloudSim.

Mapping of COs to POs:

POs/ COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PS O1	PS O2
CO1	3	3	3	2	3	–	–	–	–	1	–	3	3
CO2	3	3	3	3	3	2	–	–	–	2	–	3	3
CO3	3	2	2	2	2	–	–	–	–	1	–	2	3
CO4	3	3	2	2	3	–	–	–	–	1	–	2	3
CO5	3	3	3	3	3	–	–	–	–	2	–	3	3

Text Books:

1. Mastering Cloud Computing, 2nd edition, RajkumarBuyya, Christian Vecchiola, ThamaraiSelvi, ShivanandaPoojara, Satish N. Srirama, McGraw Hill, 2024.

2.Distributed and Cloud Computing, Kai Hwang, Geoffery C. Fox, Jack J. Dongarra, Elsevier, 2012.

3. CLOUD COMPUTING: A HANDS-ON APPROACH, Arshdeep Bahga and Vijay Madisetti, The Orient Blackswan, 2014

Reference Books:

1.Cloud Computing, Theory and Practice, Dan C Marinescu, 2nd edition, MK Elsevier, 2018.

2.Cloud Computing: Principles and Paradigms by RajkumarBuyya, James Broberg and Andrzej M. Goscinski, Wiley, 2011.

3.Online documentation and tutorials from cloud service providers (e.g. AWS, Google App Engine)

4.Docker, Reference documentation, https://docs.docker.com/reference/OpenFaaS, Serverless Functions Made Simple, https://docs.openfaas.com/

5. Mastering Cloud Computing" by Rajkumar Buyya, Christian Vecchiola, and Thamarai Selvi, Morgan Kaufmann Publishers, 2013.

III Year I Semester	CYBER SECURITY LAB	L	T	P	C
		0	0	3	1.5

Course Objective: To get practical exposure to Cybersecurity threats and Forensics tools.

Course Outcomes:

Upon the completion of the course, the students will be able to:

CO 1: Perform network reconnaissance and security analysis using port scanning, footprinting, sniffing, and honeypot deployment techniques.

CO 2: Demonstrate symmetric encryption, asymmetric encryption, hashing, digital signatures, and password generation using cryptographic tools like Jscript/Cryptool and OpenSSL.

CO 3: Apply intrusion detection and real-time traffic monitoring using Snort and perform detailed network traffic analysis using tools like Wireshark and Network Miner.

CO 4: Perform digital forensics techniques including email analysis, registry analysis, file type detection, and memory capture using Autopsy, Process Monitor, FTK Imager, and other forensic tools.

List of Experiments:

1. Perform an Experiment for port scanning with nmap
2. Set up a honeypot and monitor the honeypot on the network
3. Install Jscript/Cryptool tool (or any other equivalent) and demonstrate Asymmetric, Symmetric crypto algorithm, Hash and Digital/PKI signatures.
4. Generate minimum 10 passwords of length 12 characters using openSSL command
5. Perform practical approach to implement Footprinting - Gathering target information using Dmitry-Dmagic, UA tester
6. Work with sniffers for monitoring network communication (Wireshark).
7. Using Snort, perform real-time traffic analysis and packet logging.
8. Perform email analysis using the Autopsy tool.
9. Perform Registry analysis and get boot time logging using process monitor tool
10. Perform File type detection using Autopsy tool
11. Perform Memory capture and analysis using FTK imager tool
12. Perform Network analysis using the Network Miner tool

Mapping of COs to POs:

POs/ COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PS O1	PS O2
CO1	3	3	3	3	3	2	2	2	–	1	1	3	2
CO2	3	3	3	2	3	–	–	–	–	1	–	3	3
CO3	3	3	3	3	3	2	1	–	–	2	1	3	3
CO4	3	3	3	3	3	3	2	2	–	2	2	3	2

Text Books:

1. Real Digital Forensics for Handheld Devices, E.P. Dorothy, Auerback Publications, 2013.
2. The Basics of Digital Forensics: The Primer for Getting Started in Digital Forensics, J. Sammons, Syngress Publishing, 2012.
3. BUILD YOUR OWN CYBERSECURITY LAB :LOW-COST SOLUTIONS FOR TESTING IN VIRTUAL AND CLOUD-BASED ENVIRONMENTS, Ric Messier, McGraw Hill, 1st Edition, 2020.

Reference Books:

1. Handbook of Digital Forensics and Investigation, E. Casey, Academic Press, 2010.
2. Malware Forensics Field Guide for Windows Systems: Digital Forensics Field Guides, C.H. Malin, E. Casey and J.M. Aquilina, Syngress, 2012.
3. The Best Damn Cybercrime and Digital Forensics Book Period, J. Wiles and A. Reyes, Syngress, 2007.
4. FUNDAMENTALS OF CYBER SECURITY: Principles, Theory and Practices, Mayank Bhushan, Rajkumar Singh Rathore, Aatif Jamshed, BPB Publications; First Edition, 2017

III Year I Semester	FULLSTACK DEVELOPMENT -II	L	T	P	C
		0	1	2	2

Course Outcomes:

Upon the completion of the course, the students will be able to:

CO 1: Develop web servers, HTTP modules, user-defined modules, and implement core functionalities using Node.js.

CO 2: Apply TypeScript's advanced features including types, classes, functions, namespaces, modules, and generics for type-safe web development.

CO 3: Design interactive web pages using modern HTML5, CSS3, JavaScript, and implement responsive designs using 2D/3D transformations.

CO 4: Implement RESTful API services, session management, routing, form processing, cookies, middleware, and database connectivity using ExpressJS and MongoDB.

CO 5: Build dynamic, component-based single-page applications using ReactJS with routing, hooks, conditional rendering, state management, and event handling.

CO 6: Use MongoDB to perform CRUD operations, manage databases and collections, and handle advanced queries like indexing and aggregation.

List of Experiments:

Experiment 1: Node.js

a. Write a program to show the workflow of JavaScript code executable by creating web server in Node.js.
b. Write a program to transfer data over http protocol using http module.
c. Create a text file src.txt and add the following content to it. (HTML, CSS, Javascript, Typescript, MongoDB, Express.js, React.js, Node.js)
d. Write a program to parse an URL using URL module.
e. Write a program to create an user-defined module and show the workflow of Modularization of application using Node.js

Experiment 2: Typescript

a. Write a program to understand simple and special types.
b. Write a program to understand function parameter and return types.
c. Write a program to show the importance with Arrow function. Use optional, default and REST parameters.
d. Write a program to understand the working of typescript with class, constructor, properties, methods and access specifiers.
e. Write a program to understand the working of namespaces and modules.
f. Write a program to understand generics with variables, functions and constraints.

Experiment 3-15:

Augmented Programs: (Any 2 must be completed from **Experiment 3-5**)

3. Write a CSS program, to apply 2D and 3D transformations in a web page.
4. a web page with new features of HTML5 and CSS3.
5. Design a to-do list application using JavaScript.

Experiment 6:

ExpressJS – Routing, HTTP Methods, Middleware

a. Write a program to define a route, Handling Routes, Route Parameters, Query Parameters and URL building.
b. Write a program to accept data, retrieve data and delete a specified resource using http methods.
c. Write a program to show the working of middleware.

Experiment 7:
ExpressJS – Templating, Form Data

a. Write a program using templating engine.
b. Write a program to work with form data.

Experiment 8:
ExpressJS – Cookies, Sessions, Authentication

a. Write a program for session management using cookies and sessions.
b. Write a program for user authentication

Experiment 9:
ExpressJS – Database, RESTful APIs

a. Write a program to connect MongoDB database using Mangoose and perform CRUD operations.
b. Write a program to develop a single page application using RESTful APIs

Experiment 10:
ReactJS – Render HTML, JSX, Components – function & Class

a. Write a program to render HTML to a web page.
b. Write a program for writing markup with JSX.
c. Write a program for creating and nesting components (function and class).

Experiment 11:
ReactJS – Props and States, Styles, Respond to Events

a. Write a program to work with props and states.
b. Write a program to add styles (CSS & Sass Styling) and display data.
c. Write a program for responding to events.

Experiment 12:
ReactJS – Conditional Rendering, Rendering Lists, React Forms

a. Write a program for conditional rendering.
b. Write a program for rendering lists.
c. Write a program for working with different form fields using react forms

Experiment 13:
ReactJS – React Router, Updating the Screen

a. Write a program for routing to different pages using react router.
b. Write a program for updating the screen.

Experiment 14:
ReactJS – Hooks, Sharing data between Components

a. Write a program to understand the importance of using hooks.
b. Write a program for sharing data between components

Experiment 15:
ReactJS Applications – To-do list and Quiz

a. Design to-do list application

Experiment 16:
MongoDB – Installation, Configuration, CRUD operations

a. Install MongoDB and configure ATLAS
b. Write MongoDB queries to perform CRUD operations on document using insert(), find(), update(), remove()

Experiment 17:
MongoDB – Databases, Collections and Records
g. Write MongoDB queries to Create and drop databases and collections.
Write MongoDB queries to work with records using find(), limit(), sort(), createIndex(), aggregate()

Experiment 18-20:
Augmented Programs: (Any 2 must be completed)

18. Design a to-do list application using NodeJS and ExpressJS.
19. Design a Quiz app using ReactJS.
20. Complete the MongoDB certification from MongoDB University website.

Mapping of COs to POs:

POs/ COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PS O1	PS O2
CO1	3	2	3	2	3	–	–	–	–	1	–	3	2
CO2	3	3	3	2	3	–	–	–	–	1	–	3	2
CO3	3	2	3	1	3	–	–	–	–	2	–	3	1
CO4	3	2	3	2	3	–	–	–	–	2	–	3	3
CO5	3	2	3	2	3	–	–	–	–	2	–	3	2
CO6	3	2	3	2	3	–	–	–	–	1	–	3	3

Text Books:
1.Programming the World Wide Web, 7th Edition, Robet W Sebesta, Pearson, 2013.
2.Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node, Vasan Subramanian, 2nd edition, A Press, O’Reilly.
3.Full Stack Development with Angular and Spring Boot: Build scalable, responsive, and dynamic enterprise-level web applications, Sangeeta Joshi, BPB Publications , 2024

Web Links:
1.ExpressJS - https://www.tutorialspoint.com/expressjs
2.ReactJS - https://www.w3schools.com/REACT (and) https://react.dev/learn#
3.MongoDB - https://learn.mongodb.com/learning-paths/introduction-to-mongodb.
4. Full-Stack Web Development with FastAPI and React: Build Modern Python and JavaScript Applications. Deploy Modern Web Apps, Drake Duncan , 2025.

III Year I Semester	UI DESIGN-FLUTTER LAB	L	T	P	C
		0	0	2	1

Course Objectives:

- Learns to Implement Flutter Widgets and Layouts
- Understands Responsive UI Design and with Navigation in Flutter
- Knowledge on Widges and customize widgets for specific UI elements, Themes
- Understand to include animation apart from fetching data

Course Outcomes:

Upon the completion of the course, the students will be able to:

CO 1: . Install Flutter and Dart SDK and develop basic Dart programs to understand programming language fundamentals and Flutter architecture.

CO 2: Design mobile application UIs using Flutter widgets, layout structures, and build responsive interfaces for various screen sizes.

CO 3: Implement navigation between screens and manage application flow using both basic and named routing techniques.

CO 4: Apply state management techniques using Stateful/Stateless widgets, setState, and Provider to develop dynamic and interactive applications.

CO 5: Create custom widgets, design forms with validation, apply consistent styling using themes, and manage user input effectively.

CO 6: Integrate animations and REST API interactions to enhance user experience and display live data within the application UI.

CO 7: Perform unit testing, debugging, and code optimization to ensure the reliability and efficiency of Flutter applications.

List of Experiments:

Students need to implement the following experiments

1. a) Install Flutter and Dart SDK.
   b) Write a simple Dart program to understand the language basics.

2. a) Explore various Flutter widgets (Text, Image, Container, etc.).
   b) Implement different layout structures using Row, Column, and Stack widgets.

3. a) Design a responsive UI that adapts to different screen sizes.
   b) Implement media queries and breakpoints for responsiveness.

4. a) Set up navigation between different screens using Navigator.
   b) Implement navigation with named routes.

5. a) Learn about stateful and stateless widgets.
   b) Implement state management using set State and Provider.

6. a) Create custom widgets for specific UI elements.
   b) Apply styling using themes and custom styles.

7. a) Design a form with various input fields.
   b) Implement form validation and error handling.

8. a) Add animations to UI elements using Flutter's animation framework.
   b) Experiment with different types of animations (fade, slide, etc.).

9. a) Fetch data from a REST API.
   b) Display the fetched data in a meaningful way in the UI.

10. a) Write unit tests for UI components.
    b) Use Flutter's debugging tools to identify and fix issues.

Mapping of COs to POs:

POs/ COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PS O1	PS O2
CO1	3	2	3	2	3	–	–	–	–	1	–	3	2
CO2	3	2	3	2	3	–	–	–	–	1	–	3	2
CO3	2	2	3	2	3	–	–	–	–	1	–	3	2
CO4	3	2	3	2	3	–	–	–	–	1	–	3	2
CO5	3	2	3	2	3	–	–	–	–	1	–	3	2
CO6	3	2	3	2	3	–	–	–	–	2	–	3	3
CO7	3	2	3	3	3	–	–	–	–	2	–	3	2

Text Book:

1. Marco L. Napoli, Beginning Flutter: A Hands-on Guide to App Development.
2. Rap Payne, Beginning App Development with Flutter: Create Cross-Platform Mobile Apps 1st Edition, Apres.
3. Flutter Solutions for Web Development: Modern web development with Flutter, Dart, and AI integration, Zaid Kamil, Bpb Publications, 2025.

III Year II Semester	CYBER CRIMES & DIGITAL FORENSICS	L	T	P	C
		3	0	0	3

Course Objectives:

- Understand the threats in networks and security concepts.
- Apply authentication applications in different networks.
- Understand security services for email.
- Awareness of firewall and it applications.

Course Outcomes:

Upon the completion of the course, the students will be able to:

CO 1: Explain the fundamentals of information security, networking basics, and best practices for safe internet usage, privacy, and secure communication.

CO 2: Apply cyber ethics and laws to identify, interpret, and respond to issues involving privacy, intellectual property, freedom of speech, and cybercrimes.

CO 3: Analyze various web architectures and identify vulnerabilities through penetration testing techniques to assess security flaws in web applications.

CO 4: Evaluate web application threats such as XSS, SQL Injection, CSRF, and session hijacking; and assess forensic methods and layered defense mechanisms to ensure network assurance.

CO 5: Develop a structured information risk management and cyber incident response plan using tools like FTK or EnCase, ensuring organizational resilience and business continuity.

UNIT-I

Introduction to Information Security Fundamentals and Best Practices: Protecting Your Computer and its Contents, Securing Computer Networks--Basics of Networking, Compromised Computers, Secure Communications and Information Security Best Practices, Privacy Guidelines, Safe Internet Usage.

UNIT-II

Ethics in Cyber Security & Cyber Law: Privacy, Intellectual Property, Professional Ethics, Freedom of Speech, Fair User and Ethical Hacking, Trademarks, Internet Fraud, Electronic Evidence, Cybercrimes.

UNIT-III

Penetration Testing: Overview of the web from a penetration testers perspective, Exploring the various servers and clients, Discussion of the various web architectures, Discussion of the different types of vulnerabilities, defining a web application test scope and process, Defining types of penetration testing.

UNIT-IV

Web Application Security: Common Issues in Web Apps, what is XSS, SQL injection, CSRF, Password Vulnerabilities, SSL, CAPTCHA, Session Hijacking, Local and Remote File Inclusion, Audit Trails, Web Server Issues. **Forensics & Network Assurance:** Forensic Technologies, Digital Evidence Collection, Evidentiary Reporting, Layered Defense, Surveillance and Reconnaissance, Outsider Thread Protection

UNIT-V

Information Risk Management: Asset Evaluation and Business Impact Analysis, Risk Identification, Risk Quantification, Risk Response Development and Control, Security Policy, Compliance, and Business Continuity. Forensic investigation using Access Data FTK, En-Case
Cyber Incident Analysis and Response: Incident Preparation, Incident Detection and Analysis. Containment, Eradication, and Recovery. Proactive and Post-Incident Cyber Services, CIA triangle

Mapping of COs to POs:

POs/ COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PS O1	PS O2
CO1	3	2	2	2	3	3	2	2	–	1	–	2	2
CO2	3	2	2	–	–	3	3	3	–	1	–	2	2
CO3	3	3	3	3	3	2	–	–	–	2	–	3	3
CO4	3	2	3	3	3	3	2	2	–	2	–	3	3
CO5	3	2	3	3	2	3	3	3	–	2	–	3	3

Text Books:

1. Cyber Security & Digital Forensics by Anas Zakir, Clever Fox Publishing, Publication Date-2022
2.“Beginners Guide To Ethical Hacking and Cyber Security “, by Abhinav Ojha, Khanna Publishers, First Edition, Publication Date-2023
3. DIGITAL FORENSICS: INVESTIGATING CYBER CRIMES, Amah Nnachi Lofty , REDSHINE PUBLICATION , 2024)

Reference Books:

1.The Official CHFI Study Guide for Computer Hacking Forensic Investigator by Dave Kleiman
2.CISSP Study Guide, 6th Edition by James M. Stewart.
3. Handbook of Digital Forensics and Cyber Crime, Meenal Dhall , Dimpal, Jaisleen Kaur, Anup Kumar Kapoor, Selective & Scientific Books; First Edition, 2024

III Year II Semester	Cryptography & Network Security	L	T	P	C
		3	0	0	3

Course Objectives:

- Explain the objectives of information security
- Explain the importance and application of each of confidentiality, integrity, authentication and availability
- Understand the basic categories of threats to computers and networks
- Discusses the Mathematics of Cryptography
- Discuss the fundamental ideas of Symmetric and Asymmetric cryptographic Algorithms
- Discusses the Network layer, Transport Layer and Application layer Protocols Enhanced security mechanisms

Course Outcomes:

Upon the completion of the course, the students will be able to:

CO 1: Describe fundamental concepts of security, classical encryption techniques, and models of network security.

CO 2: Apply algebraic and number-theoretic principles to understand and solve problems related to symmetric and asymmetric cryptography.

CO 3: Analyze the structure and functioning of symmetric and asymmetric key algorithms such as DES, AES, RSA, and Elliptic Curve Cryptography.

CO 4: Evaluate the security properties and effectiveness of cryptographic hash functions, message authentication codes, and digital signature schemes.

CO 5: Design secure communication systems using protocols such as HTTPS, SSH, IPsec, S/MIME, and PGP for ensuring network and email security

(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)

UNIT I:

Security Concepts: Introduction, The need for security, Security approaches, Principles of security, Types of Security attacks, Security services, Security Mechanisms, A model for Network Security Cryptography. Classical Encryption Techniques-symmetric cipher model, Substitution techniques, Transposition techniques, Rotor Machines, Stenography.

UNIT II:

Introduction to Symmetric Cryptography: Algebraic Structures-Groups, Rings, Fields, GF(2^n) fields, Polynomials.**Mathematics of Asymmetric cryptography:** Primes, Checking For Primness, Eulers phi-functions, Fermat's Little Theorem, Euler's Theorem, Generating Primes, Primality Testing, Factorization, Chinese Remainder Theorem, Quadratic Congruence, Exponentiation And Logarithm.

UNIT III:

Symmetric key Ciphers: Block Cipher principles, DES, AES, Blowfish, IDEA, Block cipher operation, Stream ciphers: RC4, RC5. **Asymmetric key Ciphers:** Principles of public key cryptosystems, RSA algorithm, Diffie-Hellman Key Exchange, Elgamal Cryptographic system, Elliptic Curve Arithmetic, Elliptic Curve Cryptography.

UNIT IV:

Cryptographic Hash Functions: Applications of Cryptographic Hash Functions, Two Simple Hash Functions, Requirements and Security, Hash Functions Based on Cipher Block Chaining, Secure Hash Algorithms (SHA). **Message Authentication Codes:** Message Authentication Requirements, Message Authentication Functions, Requirements for Message Authentication Codes, Security of MAC'S, MAC'S Based On Hash Functions: HMAC, MAC'S Based On

Block Ciphers: DAA And CMAC. **Digital Signatures:** Digital Signatures, Elgamal Digital Signature Scheme, Elliptic Curve Digital Signature Algorithm, RSA-PSS Digital Signature Algorithm.

UNIT V:

Network and Internet Security: Transport-Level Security: Web Security Considerations, Transport Level Security, HTTPS, SSH. **IP Security:** IP Security Overview, IP Security Policy, Encapsulating Security Payload, Authentication Header Protocol. **Electronic-Mail Security:** Internet-mail Security, Email Format, Email Threats and Comprehensive Email Security, S/MIME, PGP.

Mapping of COs to POs:

POs/ COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PS O1	PS O2
CO1	3	2	2	–	–	2	2	2	–	1	–	2	2
CO2	3	3	2	–	–	–	–	–	–	–	–	3	2
CO3	3	3	3	–	2	–	–	–	–	1	–	3	2
CO4	3	3	3	–	2	2	–	–	–	2	–	3	3
CO5	3	2	3	2	3	2	–	2	–	2	–	3	3

Text Books:

1. Cryptography and Network Security - Principles and Practice: William Stallings, Pearson Education, 7th Edition, 2017
2. Cryptography and Network Security: Behrouz A. Forouzan Debdeep, Mc Graw Hill, 3rd Edition, 2015.
3. PRACTICAL MATHEMATICAL CRYPTOGRAPHY, Kristian Gjosteen, AYLOR & FRANCIS NP EXCLUSIVE(CBS); 1st edition, 2022.

Reference Books:

1. Cryptography and Network Security: Atul Kahate, Mc Graw Hill, 3rd Edition
2. Introduction to Cryptography with Coding Theory: Wade Trappe, Lawrence C. Washington, Pearson.
3. Modern Cryptography: Theory and Practice By Wenbo Mao. Pearson
4. Cryptography: Theory and Practice, Douglas Robert Stinson, Chapman and Hall/CRC; 4th edition (14 August 2018)

II Year II Semester	Machine Learning	L	T	P	C
		3	0	0	3

Course Objectives:

The objectives of the course is to

- Define machine learning and its different types (supervised and unsupervised) and understand their applications.
- Apply supervised learning algorithms including decision trees and k-nearest neighbours (k-NN).
- Implement unsupervised learning techniques, such as K-means clustering.

,

Course Outcomes:

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

CO1: Explain the fundamental concepts of machine learning, including learning paradigms, types of data, and stages in the machine learning pipeline.

CO2: Apply nearest neighbor-based classification and regression techniques using proximity and distance measures on labeled datasets.

CO3: Analyze decision tree models and probabilistic classifiers (Naive Bayes) for classification and regression, considering impurity measures and bias-variance trade-offs.

CO4: Evaluate the performance of linear models, including SVMs, logistic regression, and neural networks, for various classification and regression tasks.

CO5: Develop clustering models using algorithms like K-means, Fuzzy C-Means, Spectral Clustering, and Expectation Maximization for unsupervised data analysis.

UNIT-I Introduction to Machine Learning: Evolution of Machine Learning, Paradigms for ML, Learning by Rote, Learning by Induction, Reinforcement Learning, Types of Data, Matching, Stages in Machine Learning, Data Acquisition, Feature Engineering, Data Representation, Model Selection, Model Learning, Model Evaluation, Model Prediction, Search and Learning, Data Sets.

UNIT-II Nearest Neighbor-Based Models: Introduction to Proximity Measures, Distance Measures, Non-Metric Similarity Functions, Proximity Between Binary Patterns, Different Classification Algorithms Based on the Distance Measures ,K-Nearest Neighbor Classifier, Radius Distance Nearest Neighbor Algorithm, KNN Regression, Performance of Classifiers, Performance of Regression Algorithms.

UNIT-III Models Based on Decision Trees: Decision Trees for Classification, Impurity Measures, Properties, Regression Based on Decision Trees, Bias–Variance Trade-off, Random Forests for Classification and Regression.

The Bayes Classifier: Introduction to the Bayes Classifier, Bayes' Rule and Inference, The Bayes Classifier and its Optimality, Multi-Class Classification | Class Conditional Independence and Naive Bayes Classifier (NBC)

UNIT-IV: Linear Discriminants for Machine Learning: Introduction to Linear Discriminants, Linear Discriminants for Classification, Perceptron Classifier, Perceptron Learning Algorithm, Support Vector Machines, Linearly Non-Separable Case, Non-linear SVM, Kernel Trick, Logistic Regression, Linear Regression, Multi-Layer Perceptrons (MLPs), Backpropagation for Training an MLP.

UNIT-V: Clustering : Introduction to Clustering, Partitioning of Data, Matrix Factorization | Clustering of Patterns, Divisive Clustering, Agglomerative Clustering, Partitional Clustering, K-Means Clustering, Soft Partitioning, Soft Clustering, Fuzzy C-Means Clustering, Rough Clustering, Rough K-Means Clustering Algorithm, Expectation Maximization-Based Clustering, Spectral Clustering.

Mapping of COs to POs:

POs/ COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PS O1	PS O2
CO1	3	2	2	2	2	–	–	–	–	1	–	2	3
CO2	3	3	2	3	3	–	–	–	–	–	–	3	3
CO3	3	3	3	3	2	–	–	–	–	1	–	3	3
CO4	3	3	3	3	3	–	–	–	–	2	–	3	3
CO5	3	3	3	3	2	–	–	–	–	1	–	3	3

Text Books:

1.“Machine Learning Theory and Practice”, M N Murthy, V S Ananthanarayana, Universities Press (India), 2024.

2.Machine Learning for Real World Applications, Dinesh K. Sharma, Springer Nature; 2024th edition (9 October 2024)

Reference Books:

1.“Machine Learning”, Tom M. Mitchell, McGraw-Hill Publication, 2017
2.“Machine Learning in Action”,Peter Harrington, DreamTech
3.“Introduction to Data Mining”, Pang-Ning Tan, Michel Stenbach, Vipin Kumar, 7th Edition, 2019.
4.Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems, O'Reilly, Third Edition, 2022.

III Year **II Semester**	**SOFTWARE TESTING METHODOLOGIES** (Professional Elective-II)	**L**	**T**	**P**	**C**
		3	0	0	3

Course Objectives

- To provide knowledge of the concepts in software testing such as testing process, criteria, strategies, and methodologies.
- To develop skills in software test automation and management using the latest tools.

Course Outcomes:

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

CO1: Understand the fundamental principles of software testing including path testing, types of bugs, and flow graphs.
CO2: Apply transaction flow, data flow, and domain testing techniques to evaluate the quality and correctness of software modules.
CO3: Analyze software structure using path products, regular expressions, and logic-based testing methods to identify anomalies and improve test coverage.
CO4: Evaluate the behavior of software using state graphs and transition testing to ensure correctness and reliability in various execution conditions.
CO5: Build and interpret graph matrices, and utilize automation tools such as JMeter, Selenium, or SoapUI for efficient software testing.

UNIT - I

Introduction: Purpose of testing, Dichotomies, model for testing, consequences of bugs, taxonomy of bugs Flow graphs and Path testing: Basics concepts of path testing, predicates, path predicates and achievable paths, path sensitizing, path instrumentation, application of path testing.

UNIT - II

Transaction Flow Testing: transaction flows, transaction flow testing techniques.
Data Flow testing: Basics of data flow testing, strategies in data flow testing, application of data flow testing.
Domain Testing: domains and paths, Nice & ugly domains, domain testing, domains and interfaces testing, domain and interface testing, domains and testability.

UNIT - III

Paths, Path products and Regular expressions: path products & path expression, reduction procedure, applications, regular expressions & flow anomaly detection.
Logic Based Testing: overview, decision tables, path expressions, kv charts, specifications.

UNIT - IV

State, State Graphs and Transition testing: state graphs, good & bad state graphs, state testing, Testability tips.

UNIT - V

Graph Matrices and Application: Motivational overview, matrix of graph, relations, power of a matrix, node reduction algorithm, building tools. (Student should be given an exposure to a tool like Jmeter/selenium/soapUI/Catalon).

Mapping of COs to POs:

POs/ COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PS O1	PS O2
CO1	3	2	2	2	2	–	–	–	–	1	–	3	–
CO2	3	3	3	2	2	–	–	–	–	–	–	3	–
CO3	3	3	2	3	2	–	–	–	–	1	–	3	–
CO4	3	2	2	2	2	–	–	–	–	–	–	2	–
C05	3	3	3	3	3	–	–	–	–	2	–	3	–

Text Books:

1. Software Testing techniques - Baris Beizer, Dreamtech, second edition.
2. Software Testing Tools – Dr. K. V. K. K. Prasad, Dreamtech.
3. SOFTWARE TESTING METHODOLOGIES : EASY MADE, Dr. V. Umadevi, Notion Press; 1st edition, 2020

Reference Books:

1. The craft of software testing - Brian Marick, Pearson Education.
2. Software Testing Techniques – SPD(Oreille)
3. Software Testing in the Real World – Edward Kit, Pearson.
4. Effective methods of Software Testing, Perry, John Wiley.
5. Art of Software Testing – Meyers, John Wiley.
6. Software Testing Techniques, Boris Beizer , Wiley India; Second edition, 2002.

.

		L	T	P	C
III Year **II Semester**	**DevOps** (Professional Elective-II)	3	0	0	3

Course Objectives:
The main objectives of this course are to:

- Describe the agile relationship between development and IT operations.
- Understand the skill sets and high-functioning teams involved in DevOps and related methods to reach a continuous delivery capability.
- Implement automated system update and DevOps lifecycle.

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

CO1: Understand the fundamentals of DevOps, including SDLC models, DevOps lifecycle, architecture, and workflow principles.

CO2: Apply source code management practices using Git and perform code quality analysis with tools like SonarQube and unit testing frameworks such as JUnit or NUnit.

CO3: Develop and manage CI pipelines using Jenkins, including build automation, scheduling, and execution in distributed environments.

CO4: Analyze the concepts of Continuous Delivery and containerization using Docker and evaluate testing automation tools such as Selenium.

CO5: Implement configuration management and deployment automation using Ansible and Kubernetes, and explore tools like Puppet and Chef for orchestrating infrastructure.

UNIT-I
Introduction to DevOps: Introduction to SDLC, Agile Model. Introduction to DevOps. DevOps Features, DevOps Architecture, DevOps Lifecycle, Understanding Workflow and principles, Introduction to DevOps tools, Build Automation, Delivery Automation, Understanding Code Quality, Automation of CI/ CD. Release management, Scrum, Kanban, delivery pipeline, bottlenecks, examples

UNIT-II
Source Code Management (GIT): The need for source code control, The history of source code management, Roles and code, source code management system and migrations. What is Version Control and GIT, GIT Installation, GIT features, GIT workflow, working with remote repository, GIT commands, GIT branching, GIT staging and collaboration. UNIT TESTING - CODE COVERAGE: Junit, nUnit& Code Coverage with Sonar Qube, SonarQube - Code Quality Analysis.

UNIT-III
Build Automation - Continuous Integration (CI): Build Automation, What is CI Why Cl is Required, CI tools, Introduction to Jenkins (With Architecture), jenkins workflow, jenkins master slave architecture, Jenkins Pipelines, PIPELINE BASICS - Jenkins Master, Node, Agent, and Executor Freestyle Projects & Pipelines, Jenkins for Continuous Integration, Create and Manage Builds, User Management in Jenkins Schedule Builds, Launch Builds on Slave Nodes.

UNIT-IV
Continuous Delivery (CD): Importance of Continuous Delivery, CONTINUOUS DEPLOYMENT CD Flow, Containerization with Docker: Introduction to Docker, Docker installation, Docker commands, Images & Containers, DockerFile, Running containers, Working with containers and publish to Docker Hub.
Testing Tools: Introduction to Selenium and its features, JavaScript testing.

UNIT-V

Configuration Management - ANSIBLE: Introduction to Ansible, Ansible tasks, Roles, Jinjatemplating, Vaults, Deployments using Ansible.
CONTAINERIZATION USING KUBERNETES(OPENSHIFT): Introduction to Kubernetes Namespace & Resources, CI/CD - On OCP, BC, DC &ConfigMaps, Deploying Apps on Openshift Container Pods. Introduction to Puppet master and Chef.

Mapping of COs to POs:

POs/ COs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PS O1**	**PS O2**
CO1	3	2	2	2	–	–	–	–	–	2	–	3	3
CO2	3	3	3	2	2	–	–	–	–	2	–	3	3
CO3	3	3	3	3	3	–	–	–	–	2	–	3	3
CO4	3	3	3	3	3	–	–	–	–	2	–	3	3
CO5	3	3	3	3	3	–	–	–	–	2	–	3	3

Text Books:

1. Joyner, Joseph.,Devops for Beginners: Devops Software Development Method Guide for Software Developers and It Professionals, 1st Edition MihailsKonoplows, 2015.
2. Alisson Machado de Menezes., Hands-on DevOps with Linux,1st Edition, BPB Publications, India, 2021.
3. The DevOps Handbook: How to Create World-Class Agility, Reliability, & Security in Technology Organizations, Gene Kim, Jez Humble, Patrick Debois , John Willis, Nicole Forsgren, Shroff/IT Revolution, Second Edition, 2024.

Reference Books:

1. Len Bass, Ingo Weber, Liming Zhu. DevOps: A Software Architect's Perspective. Addison Wesley; ISBN-10
2. Gene Kim Je Humble, Patrick Debois, John Willis. The DevOps Handbook, 1st Edition, IT Revolution Press, 2016.
3. Verona, Joakim Practical DevOps, 1stEdition, Packt Publishing, 2016.
4. Joakim Verona. Practical Devops, Ingram short title; 2ndedition (2018). ISBN10: 1788392574
5. Deepak Gaikwad, Viral Thakkar. DevOps Tools from Practitioner's Viewpoint. Wiley publications. ISBN: 9788126579952.
6. Real-World DevOps Practices, B. Thangaraju, Wiley (17 October 2024).

III Year II Semester	**MICROPROCESSORS &** **MICROCONTROLLERS** (Professional Elective-II)	**L**	**T**	**P**	**C**
		3	0	0	3

Course Objectives:

- To introduce fundamental architectural concepts of microprocessors and microcontrollers.
- To impart knowledge on addressing modes and instruction set of 8086 and 8051
- To introduce assembly language programming concepts
- To explain memory and I/O interfacing with 8086 and 8051
- To introduce16 bit and 32 bit microcontrollers.

Course Outcomes:

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

CO1: Understand the architecture, internal units, system configurations, and interrupt mechanisms of the 8086 microprocessors.

CO2: Apply assembly language programming techniques using 8086 instructions, addressing modes, and development tools.

CO3: Analyze and design interfacing circuits for memory, I/O devices, and peripheral controllers like 8255, 8251, 8237, and 8259 with 8086.

CO4: Understand the architecture and instruction set of 8051 microcontroller and write assembly programs using appropriate addressing modes.

CO5: Develop embedded system applications by interfacing 8051 with peripherals such as timers, serial ports, LCDs, keyboards, sensors, and stepper motors, and compare microprocessors with microcontrollers, PIC, and ARM processors.

UNIT I

8086 Architecture: Main features, pin diagram/description, 8086 microprocessor family, internal architecture, bus interfacing unit, execution unit, interrupts and interrupt response, 8086 system timing, minimum mode and maximum mode configuration.

UNIT II

8086 Programming: Program development steps, instructions, addressing modes, assembler directives, writing simple programs with an assembler, assembly language program development tools.

UNIT III

8086 Interfacing: Semiconductor memories interfacing (RAM, ROM), Intel 8255 programmable peripheral interface, Interfacing switches and LEDS, Interfacing seven segment displays, software and hardware interrupt applications, Intel 8251 USART architecture and interfacing, Intel 8237a DMA controller, stepper motor, A/D and D/A converters, Need for 8259 programmable interrupt controllers.

UNIT IV

Microcontroller, Architecture of 8051, Special Function Registers(SFRs), I/O Pins Ports and Circuits, Instruction set, Addressing modes, Assembly language programming.

UNIT V

Interfacing Microcontroller, Programming 8051 Timers, Serial Port Programming, Interrupts Programming, LCD & Keyboard Interfacing, ADC, DAC & Sensor Interfacing, External Memory Interface, Stepper Motor and Waveform generation, Comparison of Microprocessor, Microcontroller, PIC and ARM processors

Mapping of COs to POs:

POs/ COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PS O1	PS O2
CO1	3	2	2	–	–	–	–	–	–	–	–	2	–
CO2	3	3	3	2	2	–	–	–	–	–	–	3	–
CO3	3	2	3	3	2	–	–	–	–	–	–	3	–
CO4	3	3	3	2	2	–	–	–	–	–	–	3	–
CO5	3	3	3	3	3	–	–	–	–	–	–	3	–

Textbooks:

1. Microprocessors and Interfacing – Programming and Hardware by Douglas V Hall, SSSP Rao, Tata McGraw Hill Education Private Limited, 3rd Edition,1994.
2. K M Bhurchandi, A K Ray, Advanced Microprocessors and Peripherals, 3rd edition, McGraw Hill Education, 2017.
3. Raj Kamal, Microcontrollers: Architecture, Programming, Interfacing and System Design, 2nd edition, Pearson, 2012.
4. Microprocessors & Microcontrollers, Santanu Chattopadhyay , McGraw Hill; First Edition, 2024

References:

1. Ramesh S Gaonkar, Microprocessor Architecture Programming and Applications with the 8085, 6th edition, Penram International Publishing, 2013.
2. Kenneth J. Ayala, The 8051 Microcontroller, 3rd edition, Cengage Learning, 2004.
3. Fundamentals of Microprocessors & Microcontrollers, B. Ram , Sanjay Kumar, Dhanpat Rai Publications (P) Ltd.; Latest Edition, 2021

III Year II Semester	APPLIED CRYPTOGRAPHY (Professional Elective-II)	L	T	P	C
		3	0	0	3

Course Objectives:
Knowledge on significance of cryptographic protocols and symmetric and public key algorithms

Course Outcomes:
1. Understand the various cryptographic protocols
2. Analyze key length and algorithm types and modes
3. Illustrate different public key algorithms in cryptosystems
4. Understand special algorithms for protocols and usage in the real world.

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

CO1: Understand fundamental cryptographic concepts including classical ciphers, protocol building blocks, and secure communication techniques using symmetric and asymmetric cryptography.
CO2: Analyze and compare symmetric and public key cryptographic techniques, including key lengths, algorithm types, and encryption modes.
CO3: Apply public-key algorithms and digital signature schemes such as RSA, ElGamal, and DSA for secure communication and data integrity.
CO4: Evaluate advanced cryptographic protocols and special algorithms like zero-knowledge proofs, secret sharing, blind signatures, and quantum cryptography for secure system design.
CO5: Assess the implementation of real-world cryptographic systems and standards such as Kerberos, PKCS, PGP, and UEPS in practical security infrastructures.

UNIT - I
Foundations: Terminology, Steganography, Substitution Ciphers and Transposition Ciphers, Simple XOR, One-Time Pads, Computer Algorithms, Large Numbers, Cryptographic Protocols: Protocol Building Blocks: Introduction to Protocols, Communications Using Symmetric Cryptography, One-Way Functions, One-Way Hash Functions, Communications Using Public-Key Cryptography, Digital Signatures, Digital Signatures with Encryption, Random and Pseudo-Random-Sequence Generation

UNIT - II
Cryptographic Techniques: Key length: Symmetric Key length, Public key length, comparing symmetric and public key length. Algorithm types and modes: Electronic Codebook Mode, Block Replay, Cipher Block Chaining Mode, Stream Cipher, Self-Synchronizing Stream Ciphers, Cipher-Feedback Mode, Synchronous Stream Ciphers, Output-Feedback Mod, Counter Mode, Other Block-Cipher Modes.

UNIT - III
Public-Key Algorithms: Background, Knapsack Algorithms, RSA, Pohlig-Hellman, Rabin, ElGamal, McEliece, Elliptic Curve Cryptosystems, LUC, Finite Automaton Public-Key Cryptosystems Public-Key Digital Signature Algorithms: Digital Signature Algorithm (DSA), DSA Variants, Gost Digital Signature Algorithm, Discrete Logarithm Signature Schemes, Ong-Schnorr-Shamir, ESIGN .

UNIT - IV

Special Algorithms for Protocols: Multiple-Key Public-Key Cryptography, Secret-Sharing Algorithms, Subliminal Channel, Undeniable Digital Signatures, Designated Confirmer Signatures, Computing with Encrypted Data, Fair Coin Flips, One-Way Accumulators, All-or-Nothing Disclosure of Secrets, Fair and Failsafe Cryptosystems, Zero-Knowledge Proofs of Knowledge, Blind Signatures, Oblivious Transfer, Secure Multiparty Computation, Probabilistic Encryption, Quantum Cryptography .

UNIT - V

Real World Approaches: IBM Secret key management protocol, ISDN, Kerberos, Krypto Knight, Privacy enhanced mail (PEM), Message security protocol (MSP), PGP, Public-Key Cryptography Standards (PKCS), Universal Electronic Payment System (UEPS).

Mapping of COs to POs:

POs/ COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PS O1	PS O2
CO1	3	2	–	–	2	–	–	–	–	–	–	2	3
CO2	3	3	2	–	2	–	–	–	–	–	–	2	3
CO3	3	3	3	2	3	–	–	–	–	–	–	3	3
CO4	3	3	3	2	3	–	–	–	–	–	–	3	3
CO5	3	3	3	3	3	–	–	–	–	–	–	3	3

Text Books:

1. Bruce Schneier, Applied Cryptography, Second Edition: Protocols, Algorithms, and Source Code in C (cloth).
2. Handbook of Applied Cryptography, MENEZES, TAYLOR & FRANCIS EXCL. SPL REPRINT, 2018.

Reference Book:

1. APPLIED CRYPTOGRAPHY PROTOCOLS ALGORITHM AND SOURCE CODE IN C, SCHNEIER B, JOHN WILEY (EXCLUSIVE); Standard Edition (15 May 2015)

III Year	SOFTWARE PROJECT MANAGEMENT	L	T	P	C
II Semester	(Professional Elective-III)	3	0	0	3

Course Objectives:

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

- To describe and determine the purpose and importance of project management from the perspectives of planning, tracking and completion of project
- To compare and differentiate organization structures and project structures
- To implement a project to manage project schedule, expenses and resources with the application of suitable project management tools

Course Outcomes:

At the end of the course, students will be able to

CO1: Understand conventional and modern software management practices, including software economics, process improvement, and iterative development approaches.

CO2: Analyze the various life cycle phases and process artifacts to plan and manage software development activities effectively.

CO3: Apply model-based software architecture principles, plan iterative workflows, and assess project milestones using structured checkpoints.

CO4: Evaluate project organizational structures, automation strategies, and use project control metrics for performance monitoring and process improvement.

CO5: Analyze the adoption of Agile and DevOps methodologies in software projects, including tools, team dynamics, and deployment pipelines for enhanced productivity.

UNIT-I

Conventional Software Management: The waterfall model, conventional software Management performance. **Evolution of Software Economics:** Software Economics, pragmatic software cost estimation. **Improving Software Economics:** Reducing Software product size, improving software processes, improving team effectiveness, improving automation, Achieving required quality, peer inspections. **The old way and the new:** The principles of conventional software Engineering, principles of modern software management, transitioning to an iterative process.

UNIT-II

Life cycle phases: Engineering and production stages, inception, Elaboration, construction, transition phases. **Artifacts of the process:** The artifact sets, Management artifacts, Engineering artifacts, programmatic artifacts.

UNIT- III:

Model based software architectures: A Management perspective and technical perspective. **Work Flows of the process:** Software process workflows, Iteration workflows. **Checkpoints of the process: Major** mile stones, Minor Milestones, Periodic status assessments. **Iterative Process Planning:** Work breakdown structures, planning guidelines, cost and schedule estimating, Iteration planning process, Pragmatic planning.

UNIT- IV:

Project Organizations and Responsibilities: Line-of-Business Organizations, Project Organizations, evolution of Organizations. **Process Automation:** Automation Building blocks, The Project Environment. **Project Control and Process instrumentation:** The seven core Metrics, Management indicators, quality indicators, life cycle expectations, pragmatic Software Metrics, Metrics automation.

UNIT-V:

Agile Methodology, ADAPTing to Scrum, Patterns for Adopting Scrum, Iterating towards Agility. **Fundamentals of DevOps**: Architecture, Deployments, Orchestration, Need, Instance of applications, DevOps delivery pipeline, DevOps eco system. DevOps adoption in projects: Technology aspects, Agiling capabilities, Tool stack implementation, People aspect, processes

Mapping of COs to POs:

POs/ COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PS O1	PS O2
CO1	3	2	2	–	–	–	–	–	–	–	2	2	–
CO2	3	3	3	2	–	–	–	–	–	1	2	2	–
CO3	3	3	3	2	2	–	–	–	–	–	2	3	–
CO4	3	3	3	2	2	–	–	–	1	1	3	3	–
CO5	3	3	3	2	3	–	–	–	2	2	3	3	–

Text Books:

1. Software Project Management, Walker Royce, PEA, 2005.
2. Succeeding with Agile: Software Development Using Scrum, Mike Cohn, Addison Wesley.
3. The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations, Gene Kim , John Willis , Patrick Debois , Jez Humb,1st Edition, O'Reilly publications, 2016.
4. Software Project Effort Estimation: Foundations and Best Practice Guidelines for Success, Adam Trendowicz, Springer-Nature New York Inc; Reprint edition (23 September 2016)

Reference Books:

1. Software Project Management, Bob Hughes,3/e, Mike Cotterell, TMH
2. Software Project Management, Joel Henry, PEA
3. Software Project Management in practice, PankajJalote, PEA, 2005,
4. Effective Software Project Management, Robert K.Wysocki, Wiley,2006.
5. Project Management in IT, Kathy Schwalbe, Cengage.
6. New Age Software project Management: Navigating the Technological revolution, Harshad Acharya , Adhyyan Books (12 December 2023)

III Year II Semester	MOBILE ADHOC NETWORKS (Professional Elective-III)	L	T	P	C
		3	0	0	3

Course Objectives:
From the course the student will learn

- Architect sensor networks for various application setups.
- Devise appropriate data dissemination protocols and model links cost.
- Understanding of the fundamental concepts of wireless sensor networks and has a basic knowledge of the various protocols at various layers.
- Evaluate the performance of sensor networks and identify bottlenecks.

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

CO1: Understand the characteristics, challenges, and applications of Mobile Ad Hoc Networks (MANETs), and analyze MAC protocols for ad hoc networks.
CO2: Classify and evaluate routing and transport layer protocols in Ad Hoc Wireless Networks with an emphasis on design challenges and TCP adaptations.
CO3: Identify and assess various security challenges, attacks, and secure routing techniques in Ad Hoc Wireless Networks, including intrusion detection and key management.
CO4: Analyze the architecture, communication challenges, and protocol stack of Wireless Sensor Networks (WSNs), along with their practical applications.
CO5: Apply security strategies in WSNs, understand hardware and OS platforms, and simulate sensor network protocols using tools like NS-2 and TOSSIM.

UNIT-I
Introduction to Ad Hoc Wireless Networks- Cellular and Ad Hoc Wireless Networks, Characteristics of MANETs, Applications of MANETs, Issues and Challenges of MANETs, Ad Hoc Wireless Internet, MAC protocols for Ad hoc Wireless Networks-Issues, Design Goals and Classifications of the MAC Protocols.

UNIT-II
Routing Protocols for Ad Hoc Wireless Networks- Issues in Designing a Routing Protocol, Classifications of Routing Protocols, Topology-based versus Position-based Approaches, Issues and design goals of a Transport layer protocol, Classification of Transport layer solutions, TCP over Ad hoc Wireless Networks, Solutions for TCP over Ad Hoc Wireless Networks, Other Transport layer protocols.

UNIT-III
Security protocols for Ad hoc Wireless Networks- Security in Ad hoc Wireless Networks, Network Security Requirements, Issues and Challenges in Security Provisioning, Network Security Attacks, Key Management, Secure Routing in Ad hoc Wireless Networks, Cooperation in MANETs, Intrusion Detection Systems.

UNIT-IV
Basics of Wireless Sensors and Applications- The Mica Mote, Sensing and Communication Range, Design Issues, Energy Consumption, Clustering of Sensors, Applications, Data Retrieval in Sensor Networks-Classification of WSNs, MAC layer, Routing layer, Transport layer, High-level application layer support, Adapting to the inherent dynamic nature of WSNs.

UNIT-V

Security in WSNs- Security in WSNs, Key Management in WSNs, Secure Data Aggregation in WSNs, Sensor Network Hardware-Components of Sensor Mote, Sensor Network Operating Systems–TinyOS, LA-TinyOS, SOS, RETOS, Imperative Language-nesC, **Dataflow Style Language-**TinyGALS, Node-Level Simulators, NS-2 and its sensor network extension, TOSSIM.

Mapping of COs to POs:

POs/ COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PS O1	PS O2
CO1	3	2	2	–	–	–	–	–	–	–	–	2	–
CO2	3	3	3	2	2	–	–	–	–	–	–	3	–
CO3	3	3	3	2	2	1	–	1	–	–	–	3	–
CO4	3	2	2	2	2	–	2	–	–	–	–	2	2
CO5	3	2	3	2	3	–	2	–	–	–	–	3	2

Text Books:

1. Ad Hoc Wireless Networks – Architectures and Protocols, 1st edition, C. Siva Ram Murthy, B. S. Murthy, Pearson Education, 2004
2. Ad Hoc and Sensor Networks – Theory and Applications, 2nd edition *Carlos Corderio Dharma P.Aggarwal*, World Scientific Publications / Cambridge University Press, March 2006.
3. Multimodal Biometric Security for Mobile Adhoc Network, Dr P Prabhusundhar (Author), Dr B Srinivasan, Bonfring Technology Solutions (1 January 2017).

Reference Books:

1. Wireless Sensor Networks: An Information Processing Approach, 1st edition, *Feng Zhao, Leonidas Guibas*, Elsevier Science imprint, Morgan Kauffman Publishers, 2005, rp2009
2. Wireless Ad hoc Mobile Wireless Networks – Principles, Protocols and Applications, 1st edition, Subir Kumar Sarkar, et al., Auerbach Publications, Taylor & Francis Group, 2008
3. Ad hoc Networking, 1st edition,*Charles E.Perkins*, Pearson Education, 2001
4. Wireless Ad hoc Networking, 1st edition, *Shih-Lin Wu, Yu-Chee Tseng*, Auerbach Publications, Taylor & Francis Group, 2007
5. Wireless Sensor Networks – Principles and Practice, 1st edition, Fei Hu, Xiaojun Cao, An Auerbach book, CRC Press, Taylor & Francis Group, 2010.
6. Trust Based Secure Routing in Mobile Adhoc Network, Sachi Joshi (Author), Upesh Patel, LAP Lambert Academic Publishing (26 December 2023)

III Year II Semester	NATURAL LANGUAGE PROCESSING (Professional Elective-III)	L	T	P	C
		3	0	0	3

Course Objectives:

This course introduces the fundamental concepts and techniques of natural language processing (NLP).

- Students will gain an in-depth understanding of the computational properties of natural languages and the commonly used algorithms for processing linguistic information.
- The course examines NLP models and algorithms using both the traditional symbolic and the more recent statistical approaches.
- Enable students to be capable to describe the application based on natural language processing and to show the points of syntactic, semantic and pragmatic processing.

Course Outcomes:

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

CO1: Understand the foundational concepts of NLP, including language modeling, morphological analysis, tokenization, and error correction using edit distance.

CO2: Apply and evaluate statistical techniques for word-level analysis such as N-gram models, part-of-speech tagging, and Hidden Markov Models.

CO3: Analyze syntactic structures using context-free grammars, dependency grammars, and probabilistic parsing techniques to resolve syntactic ambiguity.

CO4: Apply semantic and pragmatic analysis using logic-based representations, thematic roles, and word sense disambiguation methods for language understanding.

CO5: Evaluate discourse structures and coreference resolution techniques using lexical resources like WordNet, FrameNet, and corpora such as BNC and Penn Treebank.

UNIT I:

INTRODUCTION: Origins and challenges of NLP – Language Modeling: Grammar-based LM, Statistical LM – Regular Expressions, Finite-State Automata – English Morphology, Transducers for lexicon and rules, Tokenization, Detecting and Correcting Spelling Errors, Minimum Edit Distance.

UNIT II:

WORD LEVEL ANALYSIS: Unsmoothed N-grams, Evaluating N-grams, Smoothing, Interpolation and Backoff – Word Classes, Part- of-Speech Tagging, Rule-based, Stochastic and Transformation-based tagging, Issues in PoS tagging – Hidden Markov and Maximum Entropy models.

UNIT III:

SYNTACTIC ANALYSIS: Context-Free Grammars, Grammar rules for English, Treebanks, Normal Forms for grammar – Dependency Grammar – Syntactic Parsing, Ambiguity, Dynamic Programming parsing – Shallow parsing Probabilistic CFG, Probabilistic CYK, Probabilistic Lexicalized CFGs – Feature structures, Unification of feature structures

UNIT IV:

SEMANTICS AND PRAGMATICS: Requirements for representation, First-Order Logic, Description Logics – Syntax-Driven Semantic analysis, Semantic attachments – Word Senses, Relations between Senses, Thematic Roles, selectional restrictions – Word Sense Disambiguation, WSD using Supervised, Dictionary & Thesaurus, Bootstrapping methods – Word Similarity using Thesaurus and Distributional methods.

UNIT V:

DISCOURSE ANALYSIS AND LEXICAL RESOURCES: Discourse segmentation, Coherence – Reference Phenomena, Anaphora Resolution using Hobbs and Centering Algorithm – Coreference Resolution – Resources: Porter Stemmer, Lemmatizer, Penn Treebank, Brill's Tagger, WordNet, PropBank, FrameNet, Brown Corpus, British National Corpus (BNC).

Mapping of COs to POs:

POs/ COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PS O1	PS O2
CO1	3	2	2	–	–	–	–	–	–	–	–	2	–
CO2	3	3	2	2	2	–	–	–	–	–	–	3	2
CO3	3	2	3	2	2	–	–	–	–	–	–	3	2
CO4	3	2	2	2	2	–	–	–	–	–	–	3	2
CO5	3	2	2	2	3	–	–	–	–	–	–	3	2

Text Books:

1. Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech, 2nd Edition, Daniel Jurafsky, James H. Martin - Pearson Publication,2014.
2. Natural Language Processing with Python, First Edition, Steven Bird, Ewan Klein and Edward Loper, OReilly Media,2009.
3. Mastering Natural Language Processing with Transformers, Rupesh Kumar Tipu, LAP Lambert Academic Publishing (24 June 2024)

Reference Books:

1. Language Processing with Java and Ling Pipe Cookbook, 1st Edition, Breck Baldwin, Atlantic Publisher, 2015.
2. Natural Language Processing with Java, 2nd Edition, Richard M Reese, OReilly Media,2015.
3. Handbook of Natural Language Processing, Second, Nitin Indurkhya and Fred J. Damerau, Chapman and Hall/CRC Press, 2010.Edition
4. Natural Language Processing and Information Retrieval, 3rd Edition, Tanveer Siddiqui, U.S. Tiwary, Oxford University Press,2008.
5. Understanding Natural Language Processing, T V Geetha, Pearson Education (17 June 2024)

III Year II Semester	SECURITY ASSESSMENT AND RISK ANALYSIS (Professional Elective-III)	L	T	P	C
		3	0	0	3

COURSE OBJECTIVES

- The course takes a software development perspective to the challenges of engineering software systems that are secure.
- This course addresses design and implementation issues critical to producing secure software systems.
- The course deals with the question of how to make the requirements for confidentiality, integrity, and availability integral to the software development process.
- Secure software requirements gathering to design, development, configuration, deployment, and ongoing maintenance.
- Security of enterprise information systems.

Course Outcomes:

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

CO1: Understand the principles of secure software, threat modeling, trusted computing base, and secure deployment and management practices.

CO2: Analyze secure software design using hierarchical models, design patterns, and quality assurance strategies for early vulnerability detection.

CO3: Evaluate software assurance models, identify security risks, and apply security testing methods such as penetration testing and risk-based testing.

CO4: Apply cryptographic techniques and authentication protocols in enterprise systems, and analyze access control models, PKI, and secure communication frameworks.

CO5: Analyze security issues in the development and deployment of modern information systems including e-commerce, e-business, and e-services using standard security development frameworks.

UNIT-I

Defining computer security, the principles of secure software, trusted computing base, etc, threat modeling, advanced techniques for mapping security requirements into design specifications. Secure software implementation, deployment and ongoing management.

UNIT-II

Software design and an introduction to hierarchical design representations.Difference between high-level and detailed design.Handling security with high-level design. General Design Notions. Security concerns designs at multiple levels of abstraction, Design patterns, quality assurance activities and strategies that support early vulnerability detection, Trust models, security Architecture & design reviews .

UNIT-III

Software Assurance Model: Identify project security risks & selecting risk management strategies, Risk Management Framework, Security Best practices/ Known Security Flaws, Architectural risk analysis, Security Testing & Reliability (Penn testing, Risk- Based Security Testing

UNIT-IV

Security in Enterprise Business: Identification and authentication, Enterprise Information Security, Symmetric and asymmetric cryptography, including public key cryptography, data encryption standard (DES), advanced encryption standard (AES), algorithms for hashes and message digests. Authentication, authentication schemes , access control models, Kerberos protocol, public key infrastructure (PKI), protocols specially designed for e-commerce and web applications, firewalls and VPNs.

UNIT-V

Security development frameworks. Security issues associated with the development and deployment of information systems, including Internet-based e-commerce, e-business, and e-service systems.

Mapping of COs to POs:

POs/ COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PS O1	PS O2
CO1	3	2	2	–	2	2	–	–	–	1	–	2	–
CO2	3	3	3	2	2	–	–	–	–	–	–	3	–
CO3	3	3	3	3	2	1	–	–	–	–	–	3	–
CO4	3	2	3	2	3	–	–	–	–	–	–	3	2
CO5	3	3	2	2	2	2	–	1	–	1	–	2	2

TEXT BOOKS:

1. W. Stallings, Cryptography and network security: Principles and practice, 5 th Edition, Upper Saddle River, NJ: Prentice Hall., 2011
2. C. Kaufman, r. Perlman, & M. Speciner, Network security: Private communication in a public world, 2 nd Edition, Upper Saddle River, NJ:PrenticeHalL, 2002
3. C. P. Pfleeger, S. L. Pfleeger, Security in Computing, 4 th Edition, Upper Saddle River, NJ:Prentice Hall, 2007
4. T. M. Merkow, & J. Breithaupt, Information security: Principles and practices. Upper Saddle River, NJ:Prentice Hall, 2005.
5. Enabling Cyber Security in an Organization: Through Security Assessments and Risk Analysis, Dr. Ritu Jain Gaurav, Swashamtaa Foundation; 1.0 edition (12 October 2023)

REFERENCE BOOKS:

1. Gary McGraw, Software Security: Building Security In, Addison-Wesley, 2006.
2.Network Security Assessment: Know Your Network, Chris McNab, Shroff/O'Reilly; Third edition (1 January 2017)

III Year II Semester	Cryptography & Network Security Lab	L	T	P	C
		0	0	3	1.5

Course Objectives:

- To learn basic understanding of cryptography, how it has evolved, and some key encryption techniques used today.
- To understand and implement encryption and decryption using Ceaser Cipher, Substitution Cipher, Hill Cipher.

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

CO1: Understand and apply basic encryption and decryption techniques using classical ciphers.

CO2: Implement modern symmetric key algorithms like DES, AES (Rijndael), and Blowfish using C or Java.

CO3: Apply Java Cryptography Architecture (JCA) for secure encryption and key management.

CO4: Implement asymmetric key algorithms like RSA and key exchange protocols such as Diffie-Hellman using Java and web technologies.

CO5: Evaluate data integrity using hashing algorithms such as SHA-1 in Java.

List of Experiments:

1. Write a C program that contains a string (char pointer) with a value \Hello World’. The program should XOR each character in this string with 0 and displays the result.
2. Write a C program that contains a string (char pointer) with a value \Hello World’. The program should AND or and XOR each character in this string with 127 and display the result
3. Write a Java program to perform encryption and decryption using the following algorithms:
   a) Ceaser Cipher
   b) Substitution Cipher
   c) Hill Cipher
4. Write a Java program to implement the DES algorithm logic
5. Write a C/JAVA program to implement the BlowFish algorithm logic
6. Write a C/JAVA program to implement the Rijndael algorithm logic.
7. Using Java Cryptography, encrypt the text “Hello world” using BlowFish. Create your own key using Java key tool.
8. Write a Java program to implement RSA Algorithm
9. Implement the Diffie-Hellman Key Exchange mechanism using HTML and JavaScript. Consider the end user as one of the parties (Alice) and the JavaScript application as other party (bob).
10. Calculate the message digest of a text using the SHA-1 algorithm in JAVA.

Mapping of COs to POs:

POs/ COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PS O1	PS O2
CO1	3	2	2	–	2	–	–	–	–	–	–	2	–
CO2	3	3	3	2	3	–	–	–	–	–	–	3	–
CO3	3	3	3	2	3	–	–	–	–	–	–	3	–
CO4	3	3	3	2	3	–	–	1	–	1	–	3	–
CO5	3	3	2	2	2	–	–	–	–	–	–	3	–

III Year II Semester	Cyber Crimes & Digital Forensics Lab	L	T	P	C
		0	0	3	1.5

Course Objectives:

- Investigate cybercrime and collect evidences
- Able to use knowledge of forensic tools and software
- To understand the preservation of digital evidence.
- To learn about stenography Perceptual models

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

CO1: Apply forensic tools and techniques to collect, preserve, and analyze digital evidence from memory and storage media.
CO2: Perform live incident response, browser activity analysis, and network connection monitoring for cyber incident detection and documentation.
CO3: Demonstrate skills in digital investigation of email trails, multimedia content, and malware activity.
CO4: Analyze and evaluate forensic cases on various platforms such as Windows, Linux, networks, and mobile devices.
CO5: Identify vulnerabilities and implement forensic documentation, evidence preparation, and reporting techniques as per legal and ethical standards.

Experiment- 1

Evidence Collection

a) Linux: Capturing RAM dump using fmem
https://github.com/NateBrune/fmem
  - dcfldd if=/dev/fmem of=memory.dump hash=sha256 sha256log=memory.dump.sha256 bs=1MB count=1000

b) Linux: Capturing Disk using dfldd
https://www.obsidianforensics.com/blog/imaging-using-dcfldd
  - dcfldd if=/dev/sdb1 of=/media/disk/test_image.dd hash=md5, sha1hashlog=/media/disk/hashlog.txt

c) Windows: Capture RAM dump of a windows system
a. Hint: FTK Imager or RAMCapture
d) Windows: Capture Disk Image of a windows system
Hint: FTK Imager

Experiment- 2

Disk Analysis

i) List all files in a directory from a disk image
   a. FTK Imager
ii) Export a particular file from a disk image
   a. FTK Imager
iii) Recover a deleted file from a disk image

FTK Imager

Experiment- 3

Memory Analysis

1. List all running processes from a memory image
2. List all network connections from a memory image
3. Find out whether a firewall is set by analyzing a memory image

Hint: volatility

Experiment- 4

4) Live Incident Response
   1. Perform live incident response on a system
   2. View all browser history in a computer
   3. List out all established network connections in a computer

   Hint: Triage Incident Response

Exercise- 5

Implement E-Mail Tracking and Email Investigation

Exercise- 6

Implement video Analytics for a live video

Exercise- 7

Analysis on different Malware Working

Exercise- 8

Work on Mail Bombs &SMS bombs

Exercise- 9

Implement a case on windows and Linux forensics

Exercise- 10

Implement a case on network Forensic

Exercise- 11

Work on different types of vulnerabilities

Exercise- 12

Implement a case on Mobile Forensics

Exercise- 13

Develop a Evidence and Preparation and Documentation

Mapping of COs to POs:

POs/ COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PS O1	PS O2
CO1	3	2	2	2	3	–	–	2	–	2	–	3	2
CO2	3	3	3	3	3	2	–	2	–	2	–	3	3
CO3	3	3	2	2	3	–	–	2	–	2	–	3	2
CO4	3	2	2	3	3	–	–	2	–	2	–	3	3
CO5	2	2	2	2	2	–	–	3	–	3	3	2	2

III Year II Semester	SOFT SKILLS (Skill Enhancement Course)	L	T	P	C
		0	1	2	2

Course Objectives:

- To equip the students with the skills to effectively communicate in English
- To train the students in interview skills, group discussions and presentation skills
- To motivate the students to develop confidence
- To enhance the students' interpersonal skills
- To improve the students' writing skills

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

CO1: Understand and apply effective verbal and non-verbal communication techniques, including active listening and positive attitude development.
CO2: Demonstrate self-management skills such as stress, anger, and time management, along with social and business etiquette.
CO3: Apply fundamental English grammar and writing skills including letter writing, note making, and email communication.
CO4: Develop job-oriented skills including resume preparation, group discussions, and interview techniques through practical exercises.
CO5: Analyze interpersonal relationships, their types, and factors affecting them to foster effective collaboration and workplace harmony.

UNIT -I

Analytical Thinking & Listening Skills: Self-Introduction, Shaping Young Minds - A Talk by Azim Premji (Listening Activity), Self – Analysis, Developing Positive Attitude, Perception. Communication Skills: Verbal Communication; Non Verbal Communication (Body Language)

UNIT -II

Self-Management Skills: Anger Management, Stress Management, Time Management, Six Thinking Hats, Team Building, Leadership Qualities. Etiquette: Social Etiquette, Business Etiquette, Telephone Etiquette, Dining Etiquette

UNIT - III

Standard Operation Methods: Basic Grammars, Tenses, Prepositions, Pronunciation, Letter Writing; Note Making, Note Taking, Minutes Preparation, Email & Letter Writing

UNIT-IV

Job-Oriented Skills: Group Discussion, Mock Group Discussions, Resume Preparation, Interview Skills, Mock Interviews

UNIT-V

Interpersonal relationships: Introduction, Importance, Types, Uses, Factors affecting interpersonal relationships, Accommodating different styles, Consequences of interpersonal relationships

Mapping of COs to POs:

POs/ COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PS O1	PS O2
CO1	2	–	–	–	–	–	–	–	3	3	–	–	–
CO2	2	–	–	–	–	3	2	2	3	3	–	–	–
CO3	2	–	–	–	–	–	–	–	2	3	–	–	–
CO4	–	–	–	–	–	–	–	–	3	3	–	–	–
CO5	2	–	–	–	–	2	–	2	3	3	–	–	–

Text books:

1. Barun K. Mitra, Personality Development and Soft Skills, Oxford University Press, 2011.
2. S.P. Dhanavel, English and Soft Skills, Orient Blackswan, 2010.
3. Soft Skills, Punam Agarwal, Blue Rose Publishers (24 September 2020)

Reference books:

1. R.S.Aggarwal, A Modern Approach to Verbal & Non-Verbal Reasoning, S.Chand& Company Ltd., 2018.
2. Raman, Meenakshi& Sharma, Sangeeta, Technical Communication Principles and Practice, Oxford University Press, 2011.
3. PERSONALITY DEVELOPMENT AND SOFT SKILLS, Barun K. Mitra, Basundhara Mitra, Oxford University Press (27 February 2024)

E-resources:

https://swayam-plus.swayam2.ac.in/courses/course-details?id=P_CAMBR_01

III Year II Semester	TECHNICAL PAPER WRITING& IPR	L	T	P	C
		2	0	0	0

Course Objective:

- The course will explain the basic related to writing the technical reports and understanding the concepts related to formatting and structuring the report.
- This will help students to comprehend the concept of proofreading, proposals and practice

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

CO1: Understand the structure and elements of technical reports, sentence formation, tense usage, and planning strategies for effective technical writing.

CO2: Apply principles of drafting, editing, and layout formatting for technical documents with a focus on clarity, grammar, and plain English.

CO3: Develop skills in proofreading, summarizing, and presenting technical reports both in print and verbally, along with writing proposals.

CO4: Utilize advanced features of word processing tools such as citations, bibliographies, macros, document protection, and collaborative editing.

CO5: Understand the fundamentals of Intellectual Property Rights (IPR), including patents, copyrights, trademarks, and global cooperation in innovation and patenting processes.

UNIT-I

Introduction: An introduction to writing technical reports, technical sentences formation, using transitions to join sentences, Using tenses for technical writing. **Planning and Structuring:** Planning the report, identifying reader(s), Voice, Formatting and structuring the report, Sections of a technical report, Minutes of meeting writing.

UNIT-II

Drafting report and design issues: The use of drafts, Illustrations and graphics. **Final edits:** Grammar, spelling, readability and writing in plain English: Writing in plain English, Jargon and final layout issues, Spelling, punctuation and Grammar, Padding, Paragraphs, Ambiguity.

UNIT-III

Proofreading and summaries: Proofreading, summaries, Activities on summaries. **Presenting final reports:** Printed presentation, Verbal presentation skills, Introduction to proposals and practice.

UNIT-IV Using word processor:

Adding a Table of Contents, Updating the Table of Contents, Deleting the Table of Contents, Adding an Index, Creating an Outline, Adding Comments, Tracking Changes, Viewing Changes, Additions, and Comments, Accepting and Rejecting Changes, Working with Footnotes and Endnotes, Inserting citations and Bibliography, Comparing Documents, Combining Documents, Mark documents final and make them read only., Password protect Microsoft Word documents., Using Macros,

UNIT-V

Nature of Intellectual Property: Patents, Designs, Trade and Copyright. Process of **Patenting and Development:** technological research, innovation, patenting, development. International Scenario: International cooperation on Intellectual Property

Mapping of COs to POs:

POs/ COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PS O1	PS O2
CO1	2	–	–	–	–	–	–	–	2	3	–	–	–
CO2	2	–	–	–	–	–	–	–	2	3	–	–	–
CO3	2	–	–	–	–	–	–	–	3	3	–	–	–
CO4	2	–	–	–	3	–	–	–	2	3	–	–	–
CO5	–	–	–	–	–	2	–	3	–	2	2	–	–

Text Books:

1. Kompal Bansal & Parshit Bansal, "Fundamentals of IPR for Beginner's", 1st Ed., BS Publications, 2016.
2. William S. Pfeiffer and Kaye A. Adkins, "Technical Communication: A Practical Approach", Pearson.
3. Ramappa,T., "Intellectual Property Rights Under WTO", 2nd Ed., S Chand, 2015.
4. Technical Writing: Processes & Products, Gerson / Gerson , Pearson Education India; 8th edition (1 January 2014); Pearson India

Reference Books:

1. Adrian Wallwork , English for Writing Research Papers, Springer New York Dordrecht Heidelberg London, 2011.
2. Day R, How to Write and Publish a Scientific Paper, Cambridge University Press(2006).
3. Law Relating to Intellectual Property Rights (IPR) by MK Bhandari, Central law publication, 2024.

E-resources:

1. https://www.udemy.com/course/reportwriting/
2. https://www.udemy.com/course/professional-business-english-and-technical-report-writing/

https://www.udemy.com/course/betterbusinesswriting/