B. E . 4 Year (8 semesters) Regular Programme in INFORMATION TECHNOLOGY
Course Structure
(Applicable for the Batch admitted from the Academic Year 2023-24)

Semester V

S.NO	Code	Course Title	Scheme of Instruction				Scheme of Examination			Credits
			L	T	D/P	CONTACT HOURS	CIE	SEE	DURATION IN HOURS	Credits
Theory Courses										
1	SPC501IT	Automata Theory and Compiler Design	3			3	40	60	3	3
2	SPC502IT	Analysis and Design of Algorithms	3			3	40	60	3	3
3	SPC503IT	Internet of Things	3			3	40	60	3	3
4	SPC504IT	Software Engineering	3	-		3	40	60	3	3
5	PE-1	Professional Elective -1	3			3	40	60	3	3
Practical/Laboratory Courses										
6	SPC511IT	Internet of Things Lab			2	2	40	60	3	1
7	SPC512IT	Analysis and Design of Algorithms Lab			2	2	40	60	3	1
8	SPC513IT	Full Stack Development Lab-1 (HTML,CSS, Bootstrap, JS, ReactJS)		2	2	4	40	60	3	3
9	SPW511IT	Internship -1 (to be evaluated in 5th semester. To be carried out in summer after 4 th semester))					50	-	3	1
			15	2	6	23	370	480	27	21

1. Prof. K Syamala (University Nominee)
2. Prof. Uma N Dulhare (Subject Expert)
3. Mr. Bala Prasad P (Industry Expert)
4. Dr. Srinivasu B (Chairperson-BOS)

Professional Elective – I		
S. No.	**Course Code**	**Course Title**
1.	SPE 501 IT	Artificial Intelligence
2.	SPE 502 IT	Data Visualization And Exploration
3.	SPE 503 IT	Distributed Systems
4.	SPE 504 IT	Principles of Programming Languages
5.	SPE 505 IT	Drone Technologies

1. Prof. K Syamala (University Nominee)
2. Prof. Uma N Dulhare (Subject Expert)
3. Mr. Bala Prasad P (Industry Expert)
4. Dr. Srinivasu B (Chairperson-BOS)

Course Code	Course Title						Core/Elective
SPC501IT	**AUTOMATA THEORY AND COMPILER DESIGN**						Core
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
SPC301IT	3	-	-	-	40	60	3

Course Objectives:

1. Develop and Design finite automata to accept a set of strings of a language.
2. Understand the concepts o context free grammars and unrestricted grammars
3. To Understand and Design top-down and bottom-up parsers

Course Outcomes:

At the end of the Course the student shall be able to

1. Understand the fundamentals of formal languages and the operation of finite automata (DFA/NFA) in recognizing languages.
2. Apply regular expressions and finite automata to define and recognize patterns in regular languages.
3. Work with context-free grammars, create derivation trees, and simplify grammars using standard normal forms.
4. Understanding of Pushdown Automata and Turing Machines works and how they relate to grammars .
5. Learn how compliers works specifically lexical and syntax analysis, and use tools like LEX and YACC to implement parsers.

Additional Course Outcomes :

6.Develop and evaluate a mini-compiler or interpreter through an end-to-end case study, including lexical, syntax, and semantic analysis phases.

1. Prof. K Syamala (University Nominee)
2. Prof. Uma N Dulhare (Subject Expert)
3. Mr. Bala Prasad P (Industry Expert)
4. Dr. Srinivasu B (Chairperson-BOS)

COURSE OUTCOMES MAPPING

CO	PROGRAM OUTCOMES (PO)											PROGRAM SPECIFIC OUTCOMES (PSO)	
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	2	2	1	-	-	-	-	1	2	-	1	2	2
CO2	2	2	1	1	1	1	-	1	1	-	1	2	2
CO3	2	2	3	3	1	-	-	1	1	_	1	2	2
CO4	2	2	3	3	3	1	-	1	1		1	3	3
CO5	2	2	3	3	2	1	-	1	1		1	2	2

UNIT-I

Fundamentals: Formal Languages, Strings, Alphabets, Languages, Chomsky Hierarchy of languages.
Finite Automata: Introduction to Finite State machine, Acceptance of strings and languages, Deterministic finite automaton (DFA) and Non-deterministic finite automaton (NFA), Equivalence of NFA and DFA – Equivalence of NDFAs with and without €-moves, Minimization of finite automata, Equivalence between two DFA's, Finite automata with output – Moore and Mealy machines,

UNIT-II

Regular Languages: Regular expressions, Identity rules, Conversion of a given regular expression into a finite automaton, Conversion of finite automata into a regular expression, Pumping lemma for regular sets, Closure properties of regular sets (proofs not required).
Context Free Grammars: Context free grammars and languages, Derivation trees, Leftmost and rightmost derivation of strings and Sentential forms, Ambiguity, left recursion and left factoring in context free grammars, Minimization of context free grammars, Normal forms for context free grammars, Chomsky normal form, Greibach normal form,

UNIT-III

Pushdown Automata: Introduction to Pushdown automata, Acceptance of context free languages, Acceptance by final state and acceptance by empty state and its equivalence, Equivalence of context free grammars and pushdown automata, Inter-conversion (Proofs not required).

Turing Machine: Introduction to Turing Machine, Design of Turing machines, Types of Turing machines.

UNIT-IV

Introduction To Compiling: Overview of Compilers, Phases of a Compiler.

1. Prof. K Syamala (University Nominee)
2. Prof. Uma N Dulhare (Subject Expert)
3. Mr. Bala Prasad P (Industry Expert)
4. Dr. Srinivasu B (Chairperson-BOS)

Lexical Analysis: The Role of Lexical Analyzer, Input Buffering, Specification of Tokens, Recognition of Tokens, A language for specifying Lexical Analyzers(LEX).

UNIT-V

Syntax Analysis: The role of the Parser, First and Follow, Predictive Parsing, LR Parsers-SLR, Canonical LR, Parser Generator(YACC).

Semantic Analysis: Attributes and attribute grammars, Symbol table, Data types and Type checking, Intermediate languages, Declarations

Text Books:

1. John E Hopcroft, Rajeev Motwani, Jeffrey D.Ullman, "Introduction to Automata Theory Languages and Computation", 3rd Edition, Pearson Education, 2011.
2. Alfred Aho,Monica S Lam, RaviSethi, JeffreyD.Ullman, "Compilers- Principles Techniques and Tool", 2ndEdition, Pearson Education India, 2013.

Reference Books:

1. Peter Linz, " An introduction to Formal Languages and Automata", 6th Edition, Jones & Bartlett, 2016
2. V.Raghavan, "Principles of Compiler Design",1stEdition,McGrawHillEducation,2017.
3. Mishra and Chandrashekaran, "Theory of Computer Science – Automata Languages and Computation", 3rd Edition, PHI, 2009
4. Michel Sipser, "Introduction to Theory of Computation", 2nd Edition, Thomson, 2012

1. Prof. K Syamala (University Nominee)	2. Prof. Uma N Dulhare (Subject Expert)	3. Mr. Bala Prasad P (Industry Expert)	4. Dr. Srinivasu B (Chairperson-BOS)

Course Code	Course Title						Core/Elective
SPC502IT	**ANALYSIS AND DESIGN OF ALGORITHMS**						Core
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
SES201IT	3	-	-	-	40	60	3

Course Objectives:

This course aims to:

1. To provide an introduction to formalisms to understand, analyze and denote time complexities of algorithms.
2. To introduce the different algorithmic approaches for problem solving through numerous example problems.
3. To provide some theoretical grounding in terms of finding the lower bounds of algorithms and the NP-completeness.

Course Outcomes:

At the end of the Course the student shall be able to

1. Analyze the performance of algorithms using asymptotic notations and recurrence relations, and understand time-space trade-offs in algorithm design
2. Apply Divide and Conquer, Greedy, and Dynamic Programming techniques to solve classical algorithmic problems like matrix multiplication, knapsack, and shortest paths.
3. Understand and evaluate solutions using Backtracking and Branch-and-Bound techniques for problems such as 8-Queens, Hamiltonian Cycle, and TSP.
4. Design efficient graph-based solutions using DFS, topological sorting, and shortest path algorithms such as Dijkstra's and Bellman-Ford.
5. Understand the classes of computational complexity (P, NP, NP-Complete) and apply string matching algorithms like Boyer-More Algorithm,KMP and Rabin-Karp in practical scenarios.

Additional Course Outcomes :

6. Apply appropriate algorithm design techniques such as Divide and Conquer, Greedy, Dynamic Programming, Backtracking, and Branch and Bound to solve complex problems, and analyze their real-world applications through case studies in domains like bioinformatics, logistics, computer networks,etc.

1. Prof. K Syamala (University Nominee)
2. Prof. Uma N Dulhare (Subject Expert)
3. Mr. Bala Prasad P (Industry Expert)
4. Dr. Srinivasu B (Chairperson-BOS)

COURSE OUTCOMES MAPPING

CO	PROGRAM OUTCOMES (PO)											PROGRAM SPECIFIC OUTCOMES (PSO)	
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	3	2	2	2	-	-	-	-	-	2	3	3
CO2	3	3	3	2	2	-	-	-	1	-	1	3	3
CO3	3	3	3	3	2	-	-	-	1	-	1	3	3
CO4	3	3	3	3	3	-	--	-	1	-	1	3	3
CO5	3	3	2	2	2	-	-	-	-	-	2	3	3

UNIT - I

Introduction: Characteristics of algorithm. Analysis of algorithm: Asymptotic analysis of complexity bounds –best, average and worst-case behavior. Performance measurements of Algorithm, Time and space trade-offs.

Divide and Conquer: The general method, Minimum and Maximum Problem, Strassen's algorithm for matrix multiplication.

Analysis of recursive algorithms through recurrence relations: Substitution method, Recursion tree method and Masters' theorem.

UNIT - II

Greedy Algorithms: The general method, Knapsack Problem, Huffman Codes, Job scheduling with deadlines.

Dynamic Programming: The general method, 0/1 Knapsack, Travelling Salesman Problem, Matrix chain multiplication, Longest Common subsequence, Optimal Binary search tree.

UNIT - III

Backtracking: The general Method, 8-Queens Problem, Graph Coloring, and Hamiltonian Cycle.

Branch-and-Bound: The general method, FIFO branch and bound, LC branch and bound, 0/1 Knapsack Problem, Travelling Salesperson problem.

UNIT - IV

Graph Algorithms: Applications of DFS: Bi-Connected components, strongly connected components, topological sorting.

Shortest Path Algorithms: Dijkstra's, Bellman-Ford, Floyd-Warshall and Johnson's algorithms.

UNIT - V

1. Prof. K Syamala (University Nominee)
2. Prof. Uma N Dulhare (Subject Expert)
3. Mr. Bala Prasad P (Industry Expert)
4. Dr. Srinivasu B (Chairperson-BOS)

Theory of NP-Completeness: Polynomial time, Polynomial time verification, P, NP, NP-hard and NP-Complete classes, NP-Completeness and Reducibility.

String Matching: The naive string-matching algorithm, Boyer-More Algorithm, The Rabin-Karp algorithm, The Knuth-Morris-Pratt algorithm

Text Books:

1. Thomas H Cormen, Charles E Lieserson, Ronald L Rivest and Clifford Stein, "Introduction to Algorithms", MIT Press/McGraw-Hill, 4rd Edition, 2022.
2. E. Horowitz, sartaj sahni and sanguthevar Rajasekaran, "Fundamentals of Computer Algorithms", Uni- versities Press, 2008.

Reference Books:

1. Michael T Goodrich and Roberto Tamassia, "Algorithm Design: Foundations, Analysis", and Internet Examples, Wiley Second Edition.
2. Algorithm Design, First Edition, Jon Kleinberg and ÉvaTardos, Pearson, 2006

1. Prof. K Syamala (University Nominee)	2. Prof. Uma N Dulhare (Subject Expert)	3. Mr. Bala Prasad P (Industry Expert)	4. Dr. Srinivasu B (Chairperson-BOS)

Course Code	Course Title						Core/Elective
SPC503IT	**INTERNET OF THINGS (IoT)**						Core
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
SES302EC & SES401EC	3	-	-	-	40	60	3

Course Objectives:

1. Understand the fundamentals of IoT and its architecture.
2. Explore IoT communication models, protocols, and data handling.
3. Develop applications using Arduino, Raspberry Pi, sensors, and actuators.

Course Outcomes:

At the end of the Course the student shall be able to

1. Understand the basic concepts, characteristics, and functional components of the Internet of Things (IoT), including communication models and real-world applications.
2. Work with IoT devices like Arduino and Raspberry Pi, program and connect sensors, and understand wireless communication methods used in IoT systems like mobile networks and short-range technologies.
3. Apply IoT system design methodology and effectively interface sensors and actuators for building IoT-enabled systems.
4. Evaluate IoT protocols and service discovery mechanisms used in real-time communication and data exchange across connected devices.
5. Understand Industry 4.0 and Industrial IoT concepts, study smart application examples, and learn how to protect IoT devices using security methods like authentication, secure boot, and data encryption.

Additional Course Outcomes :

6. Study real-world IoT case studies and design secure, scalable IoT systems using new technologies like 5G, SDN, and LPWAN.

1. Prof. K Syamala (University Nominee)
2. Prof. Uma N Dulhare (Subject Expert)
3. Mr. Bala Prasad P (Industry Expert)
4. Dr. Srinivasu B (Chairperson-BOS)

COURSE OUTCOMES MAPPING

CO	PROGRAM OUTCOMES (PO)											PROGRAM SPECIFIC OUTCOMES (PSO)	
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	1	1	2	2	2	1	1	-	2	2	3
CO2	3	2	3	2	3	-	2	-	1	-	2	3	3
CO3	2	2	3	2	3	-	1	-	2	-	2	3	3
CO4	2	3	2	2	3	-	-	-	1	-	2	3	3
CO5	2	2	2	2	2	3	3	2	1	-	2	3	3

UNIT-1

Introduction: IoT–Definition, Characteristics, functional requirements, motivation, Physical design-things in IoT, IoT protocols, Logical Design-functional blocks, communication models, Communication APIs, Applications–Home Automation, Cities, Environment, Energy, Agriculture, Health, Industry.

Introduction-M2M system, Modified OSI Model for the IoT/M2M Systems, Difference between M2M and IoT, Wireless and Wired Communication Technologies, Data Enrichment, Data Consolidation And Device, Web Communication Protocols For Connected Devices, Message Communication Protocols For Connected Devices,

UNIT-II

Arduino: Introduction to Arduino Platforms, Arduino Uno architecture, IDE setup, importing Arduino boards in Arduino IDE tool, Installation of Arduino libraries.

Raspberry Pi (RPi): Basic functionality of RPi board, setting up the board by installing OS, first boot and basic configuration of Rpi,Raspberry Pi interfaces, Programming Raspberry Pi with Python

IoT Ecosystem Using Wireless Technologies : Architecture for IoT Using Mobile Devices, Mobile Technologies, 5G, Software-Defined Networking, Ultra Wide Band Technology, Near Field Communication Technology, Low Power Wide Area Networking Technologies – Sigfox, Weightless, LoRa.

UNIT-III

IoT Methodology: Purpose & Requirements specification, process, specification, domain model specification, information model specification, service specification, IoT level specifications.

1. Prof. K Syamala (University Nominee)
2. Prof. Uma N Dulhare (Subject Expert)
3. Mr. Bala Prasad P (Industry Expert)
4. Dr. Srinivasu B (Chairperson-BOS)

Sensors: Introduction, Sensor Technology, Sensor Types, Interfacing Sensors & Actuators, Communication Modules

UNIT-IV

Infrastructure and Service Discovery Protocols for the IoT Ecosystem : Layered Architecture for IoT, Protocol Architecture of IoT, IEEE 802.15.4, IPv6 over Low-Power Wireless Personal Area Networks (6LoWPAN), Bluetooth Low Energy, Long Term Evolution-Advanced, RFID, Z-Wave, Zigbee, Device or Service Discovery for IoT- Bluetooth Beacons, Wi-Fi aware, Open Hybrid.

UNIT-V

Industry 4.0: Industrial Internet of Things (IIoT), Reference Architecture, Characteristics of Industry 4.0. Case Studies: Introduction, Smart Cities, Smart Homes, Smart Lighting, Smart Transportation, Industrial Automation, Smart Healthcare, Agriculture.

Device and Data Security in IoT: Security threats and vulnerabilities in IoT devices Authentication and Authorization in IoT. Secure Boot and Firmware Updates Data integrity and confidentiality.Security protocols for constrained devices (e.g., DTLS, Lightweight Cryptography)

Text books

1. Arshdeep Bahga & Vijay Madisetti – Internet of Things: A Hands-On Approach, VPT, 2014
2. Raj Kamal – Internet of Things: Architecture and Design Principles, McGraw Hill, 2017

Reference Books:

1. Sudip Misra, Anandarup Mukherjee, Arijit Roy, “Introduction to IoT”, Cambridge University Press 2021.
2. S. Misra, C. Roy, and A. Mukherjee, 2020. Introduction to Industrial Internet of Things and Industry 4.0. CRC Press.
3. Pethuru Raj and Anupama C. Raman, "The Internet of Things: Enabling Technologies, Platforms, and Use Cases", CRC Press,2017
4. Giacomo Veneri Antonio Capasso, “Hands-On Industrial Internet of Things”, Packtr Publications, January 2018

1. Prof. K Syamala (University Nominee)	2. Prof. Uma N Dulhare (Subject Expert)	3. Mr. Bala Prasad P (Industry Expert)	4. Dr. Srinivasu B (Chairperson-BOS)

Course Code	Course Title						Core / Elective
SPC504 IT	SOFTWARE ENGINEERING						Core
Prerequisite	Contact hours per week				CIE	SEE	Credits
	L	T	D	P			
-	3	-	–	–	40	60	3

Course Objectives:

1. Understand foundational concepts of software engineering, including process frameworks, software life cycle models, and both prescriptive and agile methodologies.

2. Gain knowledge of system and requirements engineering, including techniques for requirements elicitation, modeling, and validation.

3. Develop competency in software design principles, including architectural, component-level, and user interface design, as well as the use of design patterns.

4. Acquire practical skills in software testing and debugging, applying black-box, white-box, and object-oriented testing strategies to ensure product quality.

5. Comprehend the role of software quality and metrics, including formal approaches to quality assurance, software reliability, and standards like ISO 9000.

Course Outcomes:

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

1. Acquire working knowledge of alternative approaches and techniques for each phase of software development
2. Evaluate appropriate process models by assessing software project attributes and analyzing necessary requirements for project development, eventually composing a Software Requirements Specification (SRS)
3. Create visual models to describe (non-) algorithmic solutions for projects using various design principles.
4. Develop the skills necessary to **architect** a complete software project, either independently or as part of a team, by identifying solutions for recurring problems and applying knowledge of design patterns
5. Ensure product quality through testing techniques, employing appropriate metrics and understanding the practical challenges associated with developing a significant software system.

Additional Course Outcomes : Upon successful completion of the course, students will also be able to:

1. **Critically analyze and compare software process models** (traditional vs. agile) to determine suitability for a given project context.

1. Prof. K Syamala (University Nominee)
2. Prof. Uma N Dulhare (Subject Expert)
3. Mr. Bala Prasad P (Industry Expert)
4. Dr. Srinivasu B (Chairperson-BOS)

2. **Apply system-level thinking** by integrating business process engineering and system modeling into software project planning and execution.

COURSE OUTCOMES MAPPING

CO	PROGRAM OUTCOMES (PO)											PROGRAM SPECIFIC OUTCOMES (PSO)	
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	2	-	2	1	-	-	1	-	2	2	2
CO2	2	3	3	2	2	-	-	-	1	-	2	3	2
CO3	2	2	3	2	2	-	-	-	1	-	1	2	2
CO4	2	2	3	2	2	-	-	-	3	2	2	3	2
CO5	2	3	2	2	3	2	-	-	2	-	2	3	2

UNIT-I

Software and Software Engineering : The Nature of Software, Software Engineering, The Software Process.

Process Models: A generic Process Model, Process Assessment & Improvement, The CMMI Prescriptive Process Models, Specialized Process Models, The Unified Process , Personal and Team Process Models.

Agile Development: Introduction to Agility, Agile Process, Agile Process Models.

UNIT-II

Software Engineering Principles: Core Principles, Communication Principles, Planning Principles, Modeling Principles, Construction Principles, Deployment.

System Engineering: Computer-based Systems, The System Engineering Hierarchy, Business Process Engineering, Product Engineering, System Modeling.

Requirements Engineering: A Bridge to Design and Construction, Requirements Engineering Tasks, Initiating Requirements Engineering Process, Eliciting Requirements, Developing Use-Cases, Building the Analysis Model, Negotiating Requirements, Validating Requirements.

UNIT-III

Requirements Modeling: Requirements Analysis, Scenario-Based Modeling, UML Models that supplement the use case ,Data Modeling Concepts, Class-Based Modeling.

Requirements Modeling Strategies, Flow-Oriented Modeling, Creating a Behavioral Model

Design Engineering: Design within the context of SE, Design Process, Design Concepts, The Design Model.

1. Prof. K Syamala (University Nominee)
2. Prof. Uma N Dulhare (Subject Expert)
3. Mr. Bala Prasad P (Industry Expert)
4. Dr. Srinivasu B (Chairperson-BOS)

UNIT-IV

Creating an Architectural Design: Software Architecture, Architectural Styles, Architectural Design.
Component-Level Design: Definition of Component, Designing Class-based Components, Conducting Component-level Design, Designing Traditional Components.
User Interface Design: The Golden Rules, User Interface Analysis and Design, Interface Analysis, Interface Design Steps, Design Evaluation.

UNIT-V

Software Testing Strategies: A Strategic Approach to Software Testing, Test Strategies For conventional, O-O Software, Validation Testing, System testing.
The Art of Debugging: The Debugging process, Debugging Strategies, Correcting the error.
Tactics: Software Testing Fundamentals, Black-box and White-box Testing, Basis Path Testing, Control Structure Testing, O-O Testing Methods.
Product Metrics: A Framework for Product Metrics, Metrics for each phase of software development.
Software Quality: Definition, Quality Assurance: Basic Elements, Formal Approaches, Statistical Software Quality Assurance, Software Reliability, ISO9000QualityStandards, SQA Plan.
Case Study : ***"Safe Home" Security System*** *(from Pressman's book), Online Banking System.*

Textbooks:

1. **Roger S. Pressman, Bruce R. Maxim**, Software Engineering: A Practitioner's Approach, **9th Edition**, McGraw Hill, **2023**.(e-book)
Roger S. Pressman, Bruce R. Maxim, Software Engineering: A Practitioner's Approach, **8th Edition**, McGraw Hill, **2014**.
Roger S. Pressman, Software Engineering: A Practitioner's Approach, **7th Edition**, McGraw Hill, **2009**. (refer Unit 1,2,3,4,5 from Chapters 1,2,3,4,5,6,7,8,9,10,11,16,17,18,19,23)
2. **Pankaj Jalote**, Software Engineering: With Open Source and GenAI, **2nd Edition**, Wiley India, **2024**
PankajJalote, An Integrated Approach to Software Engineering, 3rd Edition, Narosa Publishing House, 2008 (refer Unit 2 chapter 2)

Reference Books:

1. Roger S. Pressman & Bruce R. Maxim – Software Engineering: A Practitioner's Approach (9th Ed., 2023)
2. Sommerville, IanSoftware Engineering, Addison-Wesley, Boston, 9thEdition, 2011
3. Stephens, Rod Beginning Software Engineering, Wrox, 2015.

1. Prof. K Syamala (University Nominee)	2. Prof. Uma N Dulhare (Subject Expert)	3. Mr. Bala Prasad P (Industry Expert)	4. Dr. Srinivasu B (Chairperson-BOS)

Course Code	Course Title						Core / Elective
SPC 5111IT	INTERNET OF THINGS Lab						Core
Prerequisite	Contact hours per week				CIE	SEE	Credits
	L	T	D	P			
SES311EC & SES411EC		-	–	2	40	60	1

Course Objectives:

1.To become familiar with the different blocks of an IoT ecosystem.
2.To understand the working principles of Actuators
3.To be familiar with IoT devices like sensors microcontrollers, Raspberry Pi, etc.

Course Outcomes
The students will be able to:

1. Demonstrate basic programming and interfacing skills on Arduino and Raspberry Pi platforms to control sensors and actuators..
2. Interface various sensors and modules with IoT devices to capture and process environmental data.
3. Develop applications to upload sensor data to cloud platforms and design simple mobile user interfaces for IoT systems.
4. Implement wireless communication protocols such as Bluetooth and GSM in IoT applications.
5. Apply security techniques including authentication and data encryption to protect IoT devices and data communication.
6. Analyze and develop complete IoT case studies demonstrating integration of sensors, communication, cloud, mobile apps, and security features.

COURSE OUTCOMES MAPPING

CO	PROGRAM OUTCOMES (PO)											PROGRAM SPECIFIC OUTCOMES (PSO)	
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	3	2	3	-	-	-	2	-	2	3	2
CO2	2	2	2	2	3	-	-	-	1	-	2	3	3
CO3	2	2	3	2	3	-	-	-	1	-	2	3	3
CO4	2	2	3	2	3	1	-	-	1	-	2	3	3
CO5	2	2	2	2	2	3	-	2	1	-	2	3	3

1. Prof. K Syamala (University Nominee)
2. Prof. Uma N Dulhare (Subject Expert)
3. Mr. Bala Prasad P (Industry Expert)
4. Dr. Srinivasu B (Chairperson-BOS)

List of Experiments:

1. Introduction to Arduino platform and programming.
2. Controlling the Light Emitting Diode (LED) with a push button
3. Interfacing the RGB LED with the Arduino
4. Controlling the LED blink rate with the potentiometer interfacing with Arduino
5. Detection of the light using photo resistor
6. Interfacing of temperature sensor LM35 with Arduino
7. Interfacing of the Active Buzzer with Arduino.
8. Interfacing Arduino to Blue tooth Module/ GSM Module
9. Directional Control of the DC motor using Arduino
10. Setup Raspberry Pi and execute basic Python scripts
11. Interface Raspberry Pi with camera and capture image/video
12. Set up a cloud platform to upload the data from the Raspberry Pi
13. Design a Mobile App for a simple user interface
14. Implement basic authentication for IoT device communication (Arduino/RPi).
15. Encrypt sensor data transmission between Raspberry Pi and cloud.
16. Case study 1: Smart Street Light management system
17. Case study 2: Soil State Monitoring System

Text Books

1. *Embedded Controllers using C and Arduino/2E by James M. Fiore*
2. *Colin Dow, Internet of Things Programming Projects – Second Edition, Packt Publishing, 2024.*
3. *Lab Bench Studios, Programming the IoT, O'Reilly Media, 2024.*
4. *Giacomo Veneri and Antonio Capasso, Hands-on Industrial Internet of Things – Second Edition, Packt Publishing, 2024.*
5. *Simon Monk, "Programming the Raspberry Pi™ Getting Started with Python", McGraw-Hill Publications.*
6. *Arshdeep Bahga and Vijay Madisetti, Internet of Things: A Hands-On Approach, Universities Press, 2014.*
7. *Massimo Banzi and Michael Shiloh, Getting Started with Arduino, Maker Media, 2011.*

1. Prof. K Syamala (University Nominee)	2. Prof. Uma N Dulhare (Subject Expert)	3. Mr. Bala Prasad P (Industry Expert)	4. Dr. Srinivasu B (Chairperson-BOS)

Course Code	Course Title						Core / Elective
SPC512 IT	ANALYSIS AND DESIGN OF ALGORITHMS LAB						Core
Prerequisite	Contact hours per week				CIE	SEE	Credits
	L	T	D	P			
SES201IT	-	-	–	2	40	60	1

Course Objectives:

1. To understand basic concepts and structure of Algorithmic analysis.
2. To design and develop real time applications by using optimization techniques.

Course Outcomes:

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

1. Apply divide and conquer techniques to solve optimization problems such as the minimum and maximum element finding.
2. Solve optimization problems using greedy strategies including fractional knapsack and job scheduling.
3. Implement dynamic programming algorithms to solve classical problems like 0/1 Knapsack and Longest Common Subsequence.
4. Use backtracking techniques to explore feasible solutions for problems like N-Queens, Graph Coloring, and Hamiltonian Cycle.
5. Implement graph algorithms to identify bi-connected and strongly connected components, and compute shortest paths using Dijkstra's, Bellman-Ford, and Floyd-Warshall algorithms.
6. Apply pattern matching algorithms such as Boyer-Moore (BM), Knuth-Morris-Pratt (KMP), and Rabin-Karp for efficient string searching in large text data.

COURSE OUTCOMES MAPPING

CO	PROGRAM OUTCOMES (PO)											PROGRAM SPECIFIC OUTCOMES (PSO)	
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	3	2	2	2	-	-	-	-	-	2	3	3
CO2	3	3	3	2	2	-	-	-	1	-	2	3	3
CO3	3	3	3	3	2	-	-	-	1	-	2	3	3
CO4	3	3	3	3	2	-	-	-	1	-	2	3	3
CO5	3	3	3	3	2	-	-	-	1	-	2	3	3
CO6	3	3	3	3	3	-	-	-	-	-	2	3	3

1. Prof. K Syamala (University Nominee)
2. Prof. Uma N Dulhare (Subject Expert)
3. Mr. Bala Prasad P (Industry Expert)
4. Dr. Srinivasu B (Chairperson-BOS)

List of Experiments:

1. Implement problems on Divide and Conquer-Minimum-Maximum Problem
2. Implement Fractional Knapsack using greedy approach
3. Implement Job scheduling with deadlines using greedy approach
4. Implement 0/1 Knapsack using dynamic programming
5. Implement Longest Common subsequence using dynamic programming
6. Implement n-queens problem using backtracking
7. Implement graph coloring problem using backtracking
8. Implement Hamiltonian Cycle using backtracking
9. Implement bi-connected components and strongly connected components
10. Implement Dijkstra's, Bellman-Ford, Floyd-Warshall
11. Implement Boyer-Moore (BM) Algorithm.

Text Books

1. Thomas H Cormen, Charles E Lieserson, Ronald L Rivest and Clifford Stein, "Introduction to Algorithms", MIT Press/McGraw-Hill, 4rd Edition, 2022.
2. Michael T Goodrich and Roberto Tamassia, "Algorithm Design: Foundations, Analysis, and Internet Examples", Second Edition, Wiley, 2001.
3.(eBook) "Algorithm Design Techniques: Recursion, Backtracking, Greedy, Divide and Conquer, and Dynamic Programming", 2020s edition.

1. Prof. K Syamala (University Nominee)
2. Prof. Uma N Dulhare (Subject Expert)
3. Mr. Bala Prasad P (Industry Expert)
4. Dr. Srinivasu B (Chairperson-BOS)

Course Code	Course Title						Core/Elective
SPC513IT	**FULL STACK DEVELOPMENT LAB-1**						Core
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
SPC302IT& SPC412IT	-	2	-	2	40	60	3

Course Objectives:

1. To implement the static web pages using HTML and do client side validation using JavaScript.\
2. To design and work with databases using Java
3. To experiment with single page application development using React

Course Outcomes:

At the end of the Course the student shall be able to

1. Design and develop Web pages using HTML, CSS, javascript.
2. Develop Responsive web pages using frameworks.
3. Demonstrate Advanced features of JavaScript and learn about JDBC
4. Develop applications using Bootstrap, SpringBoot.
5. Develop web applications using React

COURSE OUTCOMES MAPPING

CO	PROGRAM OUTCOMES (PO)											PROGRAM SPECIFIC OUTCOMES (PSO)	
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	3	-	3	-	-	-	1	-	2	3	2
CO2	2	2	3	-	3	-	-	-	1	-	2	3	2
CO3	2	2	3	-	3	-	-	-	1	-	2	3	3
CO4	2	2	3	-	3	-	-	-	2	-	2	3	3
CO5	2	2	3	-	3	-	-	-	2	-	2	3	3

I. HTML:

1. Creation of HTML Document using basic tags.
2. Creation of Menu using ordered and unordered list and other options.
3. Creation of web page using table tags and their attributes
4. Creation of web page using frames.

1. Prof. K Syamala (University Nominee)
2. Prof. Uma N Dulhare (Subject Expert)
3. Mr. Bala Prasad P (Industry Expert)
4. Dr. Srinivasu B (Chairperson-BOS)

5. Write a HTML program to develop a static Registration Form.
6. Create a personal portfolio using basic HTML elements.
7. Write a HTML program to develop a static Web Page for Shopping Cart.

II. CSS & BOOTSTRAP

1. Write HTML for demonstration of cascading stylesheets.
    1. Embedded stylesheets.
    2. External stylesheets.
    3. Inline styles
2. Design a responsive landing page using Flexbox or Grid.
3. Design Responsive web pages using Bootstrap.

III. JAVASCRIPT & XML:

1. Basic javascript programs using control statements, arrays and functions.
2. Write a java script to validate the following fields in a registration page
    1. Name (should contains alphabets and the length should not be less than 6 characters)
    2. Password(should not be less than 6 characters)
    3. E-mail(should not contain invalid addresses)
3. Create a basic calculator using event handling.
4. Creation of XML document and validating it using DTD
5. Study and implementation of java database connectivity

V. JQery

1. Write a program to create and Build a Password Strength Check using Jquery.
2. Write a program to create and Build a star rating system using JQuery.

IV. SpringBoot Framework

1. Develop a web based application to perform Registration and Login using SpringInitilizr
2. Create a SpringBoot application to perform basic CRUD (Create/ Read / Update/ Delete) operations using MySQL.

V. REACT

1. A simple React app with increment, decrement, and reset buttons.
2. Create a signup form in React with state management and validation.
3. Manage a list of tasks using useState.
4. Fetch weather data using OpenWeatherMap API.

Software Requirements: JDK, Tomcat Server, PHP and WAMP Server.

Learning Resources:

1. Robert W. Sebesta, Programming the World Wide Web, 7th Edition (2014), Pearson Education.

1. Prof. K Syamala (University Nominee)
2. Prof. Uma N Dulhare (Subject Expert)
3. Mr. Bala Prasad P (Industry Expert)
4. Dr. Srinivasu B (Chairperson-BOS)

2. "Web Technologies", 7th Edition, Uttam K.Roy,2012.
3. Paul J. Deitel, Harvey M. Deitel, Abbey Deitel, Internet & World Wide Web How to Program, 5th Edition, Pearson Education.
4. Learning React Functional Web Development with React and Redux by Alex Banks, Eve Porcello, Alex Banks, Eve Porcello
5. Craig Walls, SpringBoot in Action, Manning Publications, 2016
6. Brad Dayley, Brendan Dayley and Caleb Dayley, Node.js, MongoDB and Angular Web Development The definitive guide to using the MEAN stackto build web applications, 2nd Edition, Pearson Education, 2018
7. http://getbootstrap.com/
8. https://reactjs.org/
9. https://nodejs.org/en/

1. Prof. K Syamala (University Nominee)

2. Prof. Uma N Dulhare (Subject Expert)

3. Mr. Bala Prasad P (Industry Expert)

4. Dr. Srinivasu B (Chairperson-BOS)

Course Code	Course Title						Core / Elective
SPE501 IT	ARTIFICIAL INTELLIGENCE						Elective
Prerequisite	Contact hours per week				CIE	SEE	Credits
	L	T	D	P			
SPC301IT	3	-	–	–	40	60	3

Course Objectives:

1. To learn basic principles of AI toward problem solving, inference, perception, knowledge representation, and learning
2. To Investigate applications of AI techniques in intelligent agents, expert systems, artificial neural networks and other machine learning models.
3. Explore the current scope, potential, limitations, and implications of intelligent systems.

Course Outcomes:

On Successful completion of the course, students will be able to

1. Understand the fundamental concepts, historical background, and components of intelligent agents and their environments in Artificial Intelligence.
2. Apply uninformed and informed search strategies, including adversarial search and constraint satisfaction methods, to solve complex AI problems.
3. Represent and reason with knowledge using semantic networks, frames, and Bayesian networks to manage uncertainty in intelligent systems.
4. Explain the design and operation of expert systems and fuzzy logic controllers, including neuro-fuzzy systems for intelligent decision-making.
5. Apply AI planning techniques such as state-space search, hierarchical planning, and multi-agent planning to real-world tasks.

Additional Course Outcomes :

6. Analyze real-world case studies to explore advanced AI applications in domains such as robotics, healthcare, smart systems, and autonomous decision-making.

1. Prof. K Syamala (University Nominee)
2. Prof. Uma N Dulhare (Subject Expert)
3. Mr. Bala Prasad P (Industry Expert)
4. Dr. Srinivasu B (Chairperson-BOS)

COURSE OUTCOMES MAPPING

CO	PROGRAM OUTCOMES (PO)											PROGRAM SPECIFIC OUTCOMES (PSO)	
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	2	-	-	-	-	-	-	-	2	2	3
CO2	3	3	3	3	3	-	-	-	1	-	2	3	3
CO3	3	3	3	3	3	-	-	-	1	-	2	3	3
CO4	3	2	3	2	3	-	-	1	1	-	2	3	3
CO5	3	3	3	3	3	-	-	-	1	-	2	3	3

UNIT-I

Introduction to Artificial Intelligence: Introduction, Brief History, Intelligent Systems, foundations of AI, Sub-Areas of AI, Applications, Tic-Tac Game Playing, Development of AI Languages, Current Trends in AI.

Agents: Agents and Environments, Good Behavior: The concept of Rationality, Performance measures, The nature of Environments, The Structure of Agents, Simple agents, Rational agents, problem solving agents, intelligent agents.

Solving problems by Search: Problem-Solving Agents, State space representation, Search graph and Search tree Searching for Solutions

Uninformed Search Strategies: Uniform cost search, Iterative deepening Depth-first search, Bidirectional search.

UNIT – II

Informed (Heuristic) Search Strategies: Heuristic Functions, Hill- climbing, Greedy best-first search, A* search, Simulated Annealing search.

Adversarial Search: Game Theory, Alpha–Beta Pruning, Constraint Satisfaction Problems.

UNIT – III

Knowledge Representation: Introduction, approaches to knowledge Representation, Knowledge Representation using Semantic Network, Knowledge Representation using Frames.

Quantifying Uncertainty- Bayes Rule -Bayesian Belief Network- Approximate Inference in Bayesian networks

UNIT - IV

Expert System: Introduction, Phases in Building Expert Systems, Expert System Architecture, Expert System versus Traditional Systems, Rule-Based Expert Systems.

Fuggy Logic Systems: Introduction, Crisp Sets, Fuzzy Sets, Fuzzy Terminology, Fuggy Logic Control,

1. Prof. K Syamala (University Nominee)
2. Prof. Uma N Dulhare (Subject Expert)
3. Mr. Bala Prasad P (Industry Expert)
4. Dr. Srinivasu B (Chairperson-BOS)

Neuro Fuzzy Systems.

UNIT – V

Classical planning, Planning as State-space search, Forward search, backward search, Planning graphs, Hierarchical Planning, Planning and acting in Nondeterministic domains — Sensor-less Planning, Mul\iagent planning

Text Book

1. **Stuart Russell and Peter Norvig**, *Artificial Intelligence: A Modern Approach*, 4th Edition, Pearson, 2020.
2. **2. K. R. Chowdhary**, *Fundamentals of Artificial Intelligence*, Springer, 2020.
3. Elaine Rich, Kevin Knight, Shivashankar B. Nair, *Artificial Intelligence*, 3rd Edition, McGraw-Hill Education, 2021.
4.Nils J. Nilsson, *The Quest for Artificial Intelligence: A History of Ideas and Achievements*, Cambridge University Press, 2010.

1. Prof. K Syamala (University Nominee)
2. Prof. Uma N Dulhare (Subject Expert)
3. Mr. Bala Prasad P (Industry Expert)
4. Dr. Srinivasu B (Chairperson-BOS)

Course Code	Course Title				Core / Elective		
SPE502 IT	DATA VISUALIZATION AND EXPLORATION				Elective		
Prerequisite	Contact hours per week				CIE	SEE	Credits
	L	T	D	P			
SPC412IT	3	-	–	–	40	60	3

Course Objectives:

1. To learn basic principles of AI toward problem solving, inference, perception, knowledge representation, and learning
2. To Investigate applications of AI techniques in intelligent agents, expert systems, artificial neural networks and other machine learning models.
3. Explore the current scope, potential, limitations, and implications of intelligent systems.

Course Outcomes:

On Successful completion of the course, students will be able to

1. Explain the role of agents and interaction with the environment to establish goals.
2. Identify and formulate search strategies to solve problems by applying suitable search strategy.
3. Compare and contrast the various knowledge representation schemes of AI.
4. Appraise probabilistic reasoning and model building
5. Apply Markov decision Process to solve real world Problems.

COURSE OUTCOMES MAPPING

CO	PROGRAM OUTCOMES (PO)											PROGRAM SPECIFIC OUTCOMES (PSO)	
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	2	-	-	-	-	-	-	-	2	2	3
CO2	3	3	3	2	3	-	-	-	1	-	2	3	3
CO3	3	2	2	2	2	-	-	-	-	-	2	3	3
CO4	3	3	3	2	3	-	-	-	1	-	2	3	3
CO5	3	3	3	3	3	-	-	-	1	-	2	3	3

1. Prof. K Syamala (University Nominee)
2. Prof. Uma N Dulhare (Subject Expert)
3. Mr. Bala Prasad P (Industry Expert)
4. Dr. Srinivasu B (Chairperson-BOS)

UNIT I: Foundations of Data Exploration and Preparation: Introduction to Data Exploration and Visualization: Goals, Importance, and the Data Science Pipeline, Types of Data and Their Characteristics, Introduction to Common Visualization Tools and Libraries. Data Preparation and Preprocessing: Handling Missing Values, Outliers, Duplicates.
Data Transformation: Scaling, Normalization, Encoding Categorical Variables.
Introduction to Python Libraries for Data Manipulation (Pandas basics).

UNIT II: Exploratory Data Analysis (EDA) and Basic Visualization

Univariate Analysis: Visualizing Distributions (Histograms, Density Plots, Box Plots) using Matplotlib. Bivariate Analysis: Exploring Relationships between Two Variables (Scatter Plots, Line Plots) using Matplotlib.
Customizing Basic Visualizations: Labels, Titles, Legends, Colors. Group-wise Analysis and Aggregations using Pandas and basic plotting. Identifying Initial Patterns and Trends through Basic Visualizations.

UNIT III: Principles of Effective Visualization and Advanced Techniques

Principles of Effective Data Visualization: Visual Perception and Cognition, Choosing the Right Chart Type.
Advanced Visualization Techniques: Bar Charts (grouped, stacked), Pie Charts (and their limitations), Box Plots, Violin Plots using Seaborn.Heatmaps and Correlation Matrices for Multivariate Analysis. Introduction to Statistical Visualization with Seaborn. Avoiding Misleading Visualizations and Common Pitfalls.

UNIT IV: Multivariate and Interactive Visualization

Multivariate Analysis: Visualizing Relationships among Multiple Variables (Pair Plots, Parallel Coordinates - introduction). Time Series Visualization (basic techniques and considerations). Introduction to Interactive Data Visualization Concepts and Tools (e.g., Plotly, Bokeh basics). Creating Basic Interactive Plots for Exploration.

UNIT V: Applications, Tools, and Evaluation

Geospatial Visualization (Introduction). Visualization for Specific Data Types (Categorical, Numerical, Text). Overview of Various Data Exploration and Visualization Tools (revisiting and expanding on different software).
Case Studies and Real-world Applications of Data Exploration and Visualization. Evaluating and Critiquing Visualizations.

1. Prof. K Syamala (University Nominee)	2. Prof. Uma N Dulhare (Subject Expert)	3. Mr. Bala Prasad P (Industry Expert)	4. Dr. Srinivasu B (Chairperson-BOS)

Text Books:

1. "Python for Data Analysis" by Wes McKinney (2017, 2nd Edition).
2. "Python Data Science Handbook" by Jake VanderPlas (2016).
3. "Storytelling with Data: A Data Visualization Guide for Business Professionals" by Cole Nussbaumer Knaflic (2015).
4. "The Visual Display of Quantitative Information" by Edward R. Tufte (2001, 2nd Edition).
5. "Information Visualization: Perception for Design" by Colin Ware (2012, 3rd Edition).
6. “Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow" by Aurélien Géron (2019, 2nd Edition).

1. Prof. K Syamala (University Nominee)	2. Prof. Uma N Dulhare (Subject Expert)	3. Mr. Bala Prasad P (Industry Expert)	4. Dr. Srinivasu B (Chairperson-BOS)

Course Code	Course Title						Core / Elective
SPE503 IT	DISTRIBUTED SYSTEMS						Elective
Prerequisite	Contact hours per week				CIE	SEE	Credits
	L	T	D	P			
SPC401IT & SPC303IT	3	-	–	–	40	60	3

Course Objectives:

1. To understand what is a distributed system and IPC, Group Communication & RPC Concepts.
2. To understand the DFS and different Name Services, virtual time, agreement and consensus protocols.
3. To understand the concepts of replication and transaction in distributed environments and associated concepts, namely, concurrency control, deadlocks and error recovery.

Course Outcomes:

On Successful completion of the course, students will be able to.

1. Understand the concepts, challenges of distributed systems and various system models.
2. Analyze the establishment of Inter process communication and remote invocation between distributed systems.
3. Comprehend a distributed system with the features that support distributed file systems and name services.
4. Relate virtual time, agreement and consensus protocols in distributed systems.
5. Apply and analyze the knowledge of distributed transactions and replication.

COURSE OUTCOMES MAPPING

CO	PROGRAM OUTCOMES (PO)											PROGRAM SPECIFIC OUTCOMES (PSO)	
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	2	-	2	-	-	-	-	-	2	3	2
CO2	3	3	3	2	3	-	-	-	1	-	2	3	3
CO3	3	2	3	2	3	-	-	-	1	-	2	3	2
CO4	3	3	3	2	3	-	-	-	1	-	2	3	3
CO5	3	3	3	2	3	-	-	-	1	-	2	3	3

1. Prof. K Syamala (University Nominee)
2. Prof. Uma N Dulhare (Subject Expert)
3. Mr. Bala Prasad P (Industry Expert)
4. Dr. Srinivasu B (Chairperson-BOS)

UNIT 1:

Characterization of Distributed Systems- Introduction, Examples of Distributed systems, Trends in distributed Systems, Focus on resource sharing, Challenges, Case study: The World Wide Web.
System models- Introduction, Physical models, Architectural models, Fundamental models.

UNIT 2:
Inter Process Communication- Introduction, The API for the Internet Protocols, External data representation and Marshalling, Multicast communication.
Remote Invocation- Introduction, Request-reply protocols, Remote Procedure Call, Remote method invocation, Case study: Java RMI.

UNIT 3:

Distributed File Systems- Introduction, File Service architecture, Case study: SUN network file system.Name Services- Introduction, Name Services and the Domain Name System, Directory Services, Case study: The Global Name Service.

UNIT 4:
Time and Global States- Introduction, Synchronizing physical clocks, Logical time and logical clocks, Global states.
Coordination and Agreement- Introduction, Distributed mutual exclusion, Elections, Coordination and agreement in group communication, Consensus and related problems.

UNIT 5:
Distributed Transactions- Introduction, Flat and Nested Distributed Transactions, Atomic commit protocols, Concurrency control in distributed transactions, Distributed deadlocks, Transaction recovery.
Replication- Introduction, System model and the role of group communication, Fault-tolerant services, Case study: The gossip architecture.

Text Books:
1. George Coulouris, J Dollimore, Tim Kindberg and G Blair, Distributed Systems, Concepts and Design, 5th Edition, Pearson Education, 2012.

Reference Books:
1. Andrew S.Tanenbaum, Maarten VanSteen, Distributed systems, Principles and Paradigms, 2nd Edition, PHI.
2. Sukumar Ghosh, Chapman & Hall/CRC, Taylor & Fransis Group, Distributed Systems, An Algorithm Approach, 2007.

1. Prof. K Syamala (University Nominee)	2. Prof. Uma N Dulhare (Subject Expert)	3. Mr. Bala Prasad P (Industry Expert)	4. Dr. Srinivasu B (Chairperson-BOS)

Course Code	Course Title				Core / Elective		
SPE504 IT	PRINCIPLES OF PROGRAMMING LANGUAGES				Elective		
Prerequisite	Contact hours per week				CIE	SEE	Credits
	L	T	D	P			
SPS101IT	3	-	–	–	40	60	3

Course Objectives:

1. To introduce the various programming paradigms.
2. To understand the evolution of programming languages and concepts of OO languages, functional languages, logical, and scripting languages.
3. To introduce the principles and techniques involved in the design and implementation of modern programming languages. concurrency control and exception handling, ADT and OOP for software development.

Course Outcomes:

On Successful completion of the course, students will be able to.

1. Express syntax and semantics in formal notation.
2. Apply a suitable programming paradigm for the application
3. Compare the features of various programming languages.
4. Understand the programming paradigms of modern programming languages.
5. Understand the concepts of ADT and OOP.

Additional Course Outcomes :

6. Correlate different language paradigms and evaluate their relative benefits.

COURSE OUTCOMES MAPPING

CO	PROGRAM OUTCOMES (PO)											PROGRAM SPECIFIC OUTCOMES (PSO)	
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	2	2	2	-	-	-	-	-	2	3	2
CO2	3	3	3	2	3	-	-	-	1	-	2	3	3
CO3	3	2	2	1	2	-	-	-	-	-	2	2	2
CO4	3	2	2	1	2	-	-	-	-	-	2	2	2
CO5	3	2	3	1	3	-	-	-	1	-	2	3	2

1. Prof. K Syamala (University Nominee)
2. Prof. Uma N Dulhare (Subject Expert)
3. Mr. Bala Prasad P (Industry Expert)
4. Dr. Srinivasu B (Chairperson-BOS)

UNIT I

Preliminary Concepts: Reasons for Studying Concepts of Programming Languages, Programming Domains, Language Evaluation Criteria, Influences on Language Design, Language Categories, Language Design Trade-Offs, Implementation Methods,
Programming Environments Syntax and Semantics: General Problem of Describing Syntax and Semantics, Formal Methods of Describing Syntax, Attribute Grammars, Describing the Meanings of Programs.

UNIT II

Names, Bindings, and Scopes-Introduction, Names, Variables, Concept of Binding, Scope, Scope and Lifetime, Referencing Environments, Named Constants Data Types-Introduction, Primitive Data Types, Character String Types, User Defined Ordinal Types, Array, Associative Arrays, Record, Union, Tuple Types, List Types, Pointer and Reference Types, Type Checking, Strong Typing, Type Equivalence Expressions and Statements-Arithmetic Expressions, Overloaded Operators, Type Conversions, Relational and Boolean Expressions, Short Circuit Evaluation, Assignment Statements, Mixed-Mode Assignment Control Structures –Introduction, Selection Statements, Iterative Statements, Unconditional Branching, Guarded Commands.

UNIT III

Subprograms and Blocks- Fundamentals of Sub-Programs, Design Issues for Subprograms, Local Referencing Environments, Parameter Passing Methods, Parameters that Are Subprograms, Calling Subprograms Indirectly,Overloaded Subprograms, Generic Subprograms, Design Issues for Functions, User Defined Overloaded Operators, Closures, Coroutines

Implementing Subprograms- General Semantics of Calls and Returns, Implementing Simple Subprograms, Implementing Subprograms with Stack-Dynamic Local Variables, Nested Subprograms, Blocks, Implementing Dynamic Scoping.

Abstract Data Types- The Concept of Abstraction, Introductions to Data Abstraction, Design Issues, Language Examples, Parameterized ADT, Encapsulation Constructs, Naming Encapsulations.

UNIT IV

Concurrency- Introduction, Introduction to Subprogram Level Concurrency, Semaphores, Monitors, Message Passing, Java Threads, Concurrency in Function Languages, Statement Level Concurrency.
Exception Handling and Event Handling: Introduction, Exception Handling in Ada, C++, Java, Introduction to Event Handling, Event Handling with Java and C#.

1. Prof. K Syamala (University Nominee)
2. Prof. Uma N Dulhare (Subject Expert)
3. Mr. Bala Prasad P (Industry Expert)
4. Dr. Srinivasu B (Chairperson-BOS)

UNIT V

Functional Programming Languages- Introduction, Mathematical Functions, Fundamentals of Functional Programming Language, LISP, Support for Functional Programming in Primarily Imperative Languages, Comparison of Functional and Imperative Languages Logic Programming Language- Introduction, an Overview of Logic Programming, Basic Elements of Prolog, Applications of Logic Programming.

Scripting Language- Pragmatics, Key Concepts, Case Study: Python – Values and Types, Variables, Storage and Control, Bindings and Scope, Procedural Abstraction, Data Abstraction, Separate Compilation.

Text Books:

1. Robert .W. Sebesta, Concepts of Programming Languages, 10th Edition, Pearson Education.
2. D. A. Watt, Programming Language Design Concepts, Wiley India Edition.

Reference Books:

1. A.B. Tucker, R.E. Noonan, Programming Languages, TMH.
2. K. C. Louden and K A Lambert., Programming Languages, 3rd edition, Cengage Learning.
3. C Ghezzi and M Jazayeri, Programming Language Concepts, Wiley India.
4. Ravi Sethi, Programming Languages, 2nd Edition, Pearson.

1. Prof. K Syamala (University Nominee)
2. Prof. Uma N Dulhare (Subject Expert)
3. Mr. Bala Prasad P (Industry Expert)
4. Dr. Srinivasu B (Chairperson-BOS)

Course Code	Course Title						Core / Elective
SPE505 IT	DRONE TECHNOLOGY						Elective
Prerequisite	Contact hours per week				CIE	SEE	Credits
	L	T	D	P			
SES401EC	3	-	–	–	40	60	3

Course Objectives:

1.Learn the basics of drones, their parts, and how they are used in different fields.

2.Understand how drones fly and how to control their movements using sensors and software.

3.Know the rules and safety measures for flying drones and explore new drone technologies.

Course Outcomes:

1. Describe various types of drones, their components, and real-world applications.
2. Apply principles of aerodynamics and flight mechanics to analyze drone stability and control.
3. Design and troubleshoot drone control systems using sensors and communication technologies.
4. Implement and simulate autonomous drone missions using standard drone software platforms.
5. Evaluate drone regulations and ethical considerations to ensure safe and legal UAV operation.

Additional Course Outcomes :

6. Analyze emerging drone technologies and perform case studies on innovative applications.*(For advanced learners)*

COURSE OUTCOMES MAPPING

CO	PROGRAM OUTCOMES (PO)											PROGRAM SPECIFIC OUTCOMES (PSO)	
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	2	2	1	-	-	-	-	1	2	-	1	2	2
CO2	2	2	1	1	1	1	-	1	1	-	1	2	2
CO3	2	2	3	3	1	-	-	1	1	_	1	2	2
CO4	2	2	3	3	3	1	-	1	1		1	3	3
CO5	2	2	3	3	2	1	-	1	1		1	2	2

1. Prof. K Syamala (University Nominee)
2. Prof. Uma N Dulhare (Subject Expert)
3. Mr. Bala Prasad P (Industry Expert)
4. Dr. Srinivasu B (Chairperson-BOS)

UNIT-I

Introduction to Drones and UAV Systems: History and evolution of drones,Classification and types of UAVs (fixed-wing, rotary-wing, hybrid), Components of drone systems: frame, motors, ESC, propellers, flight controller, sensors,Applications of drones in agriculture, defense, delivery, surveying, and photography

UNIT-II

Aerodynamics and Flight Mechanics: Basic principles of flight and aerodynamics for drones Lift, drag, thrust, and weight forces,Stability and control of drones,Flight dynamics and modeling of quadcopters and other UAV types.

UNIT-III

Drone Electronics and Control Systems: Flight controllers: architecture and functioning

Sensors for navigation and stabilization: IMU, GPS, barometer, magnetometer

Communication protocols: RC, telemetry, radio control systems,Introduction to autopilot systems and PID control.

UNIT-IV

Drone Software and Programming: Introduction to drone programming platforms (e.g., ArduPilot, PX4), Mission planning and autonomous flight,Integration with ground control stations (GCS),Basics of drone simulation environments.

UNIT-V

Regulations, Safety, and Emerging Trends: Drone laws, regulations, and certifications worldwide (FAA, DGCA, EASA),Privacy, ethical considerations, and safety standards,Emerging technologies: swarm drones, AI-based drones, drone delivery systems,Case studies on drone applications

Text Book:

1. Prof. K Syamala (University Nominee)	2. Prof. Uma N Dulhare (Subject Expert)	3. Mr. Bala Prasad P (Industry Expert)	4. Dr. Srinivasu B (Chairperson-BOS)

1.Beard, R. W., & McLain, T. W. (2019). *Small Unmanned Aircraft: Theory and Practice* (2nd ed.). Princeton University Press.

2.Anderson, D. V., & Hollett, R. C. (2021). *Introduction to Unmanned Aircraft Systems* (3rd ed.). Wiley.

3.Austin, R. (2010). *Unmanned Aircraft Systems: UAVS Design, Development and Deployment* (2nd ed.). Wiley.

4.Holland, J. M. (2021). *Introduction to UAV Systems* (4th ed.). Wiley.

5..Valavanis, K. P., & Vachtsevanos, G. J. (Eds.). (2015). *Handbook of Unmanned Aerial Vehicles*. Springer.

1. Prof. K Syamala (University Nominee)

2. Prof. Uma N Dulhare (Subject Expert)

3. Mr. Bala Prasad P (Industry Expert)

4. Dr. Srinivasu B (Chairperson-BOS)