CURRICULUM & SYLLABI

Master of Computer Applications(MCA)

Effective from AY: 2025-26



NATIONAL INSTITUTE OF TECHNOLOGY WARANGAL

WARANGAL, TELANGANA

Table of Contents

VISION AND MISSION OF THE INSTITUTE 3
VISION AND MISSION OF THE DEPARTMENT 3
PROGRAM EDUCATIONAL OBJECTIVES 4
PROGRAM OUTCOMES 5
CURRICULUM 6
SYLLABI 12

Vision and Mission of the Institute

National Institute of Technology Warangal

VISION

Towards a Global Knowledge Hub, striving continuously in pursuit of excellence in Education, Research, Entrepreneurship and Technological services to the society.

MISSION

- Imparting total quality education to develop innovative, entrepreneurial and ethical future professionals fit for globally competitive environment.
- Allowing stake holders to share our reservoir of experience in education and knowledge for mutual enrichment in the field of technical education.
- Fostering product-oriented research for establishing a self-sustaining and wealth creating centre to serve the societal needs.

Vision and Mission of the Department

Computational & Data Science

VISION

To build an international reputation as a center of excellence in Computational & Data Science research, training, and education, and to inculcate scientific thinking to meet the challenges and growth of science and technology as well as the needs of industry and society, with moral and ethical responsibility.

MISSION

- Imparting quality education through well-designed curriculum in tune with the challenging formal models and software system needs of the industry.
- Providing state-of-art research facilities to generate knowledge and develop technologies in the thrust areas of Computational and Data Science.
- Developing linkages with world class organizations to strengthen industry-academia relationships for mutual benefit

Program: Master of Computer Applications

Program Educational Objectives

PEO-1	Design applications for real-world problems and analyze their Complexities by testing.
PEO-2	Design and develop user interface frameworks among the subsystems to enhance portability.
PEO-3	Discover knowledge from large data sets to analyze technical solutions for complexapplications.
PEO-4	Work in teams to learn and assess security, privacy, cost and quality assurance in developing software systems.
PEO-5	Engage in lifelong learning to keep pace with changing landscape of technologies forprofessional advancement.

Program Articulation Matrix

PEO / Mission Statements	PEO-1	PEO-2	PEO-3	PEO-4	PEO-5
Imparting quality education through well-designed curriculum in tune with the challenging software needs of the industry.	3	3	2	3	2
Providing state-of-art research facilities to generate knowledge and develop technologies in the thrust areas of Computer Science and Engineering.	2	3	3	3	2

Developing linkages with world class organizations to strengthen industry-academia relationships for mutual benefit	2	3	2	3	2

1 - Slightly; 2 - Moderately; 3 - Substantially

Program: Master of Computer Applications

Program Outcomes

PO-1	Apply mathematical foundation and domain knowledge for conceptualizing the computing models for real-life problems.
PO-2	Design solutions for complex business scenarios or processes that meet the specific needs of societal problems.
PO-3	Infer and predict the knowledge from data and provide synthesis to derive valid conclusions.
PO-4	Create, select, and apply appropriate techniques, resources, modern computing tools, and skills required for innovative software solutions.
PO-5	Ability to design, develop, deploy and manage robust and reliable software projects by satisfying the realistic, economic, social, safety and security constraints.
PO-6	Demonstrate a higher level of professional skills to communicate with the peer fraternity by effectively presenting the reports on complex activities.

CURRICULUM
Master of Computer Applications (MCA)

1st Semester

S.No.	Code	Course Title	L-T-P	Credits
1	MA16061	Statistics and Queuing Theory	3-0-0	3
2	CD16001	Problem Solving through Programming	4-0-0	4
3	CD16003	Computer Organization	3-0-0	3
4	CD16005	Discrete Mathematics	3-0-0	3
5	CD16007	Programming Lab	0-1-2	2
6	CD16009	Web Programming	3-0-0	3
7	CD16011	Web Programming Lab	0-1-2	2
			Total Credits	**20**

2nd Semester

S.No.	Code	Course Title	L-T-P	Credits
1	MS16064	Management Theory and Practice	3-0-0	3
2	CD16002	Software Engineering	3-0-2	4
3	CD16004	Data Structures	3-0-0	3
4	CD16006	Object Oriented Programming	3-0-0	3
5	CD16008	Algorithm Design	3-0-0	3
6	CD16010	Data Structures Lab	0-1-2	2
7	CD16012	Object Oriented Programming Lab	0-1-2	2
			Total Credits	**20**

3rd Semester

S.No.	Code	Course Title	L-T-P	Credits
1	CD17001	Python Programming	3-0-2	4
2	CD17003	Operating Systems Principles	3-0-0	3
3	CD17005	Principles of Database Management Systems	3-0-0	3
4	CD170XX	Professional Elective – I	3-0-0	3
5	CD170XX	Professional Elective – II	3-0-0	3
6	CD17007	Database Management Systems Lab	0-1-2	2
7	CD17009	Operating Systems Lab	0-1-2	2
			Total Credits	**20**

4th Semester

S.No.	Code	Course Title	L-T-P	Credits
1	CD17002	Software Testing	3-0-2	4
2	CD17004	Computer Networks	3-0-0	3
3	CD17006	Machine Learning	3-0-0	3
4	CD170XX	Professional Elective – III	3-0-0	3
5	CD170XX	Professional Elective – IV	3-0-0	3
6	CD17008	Computer Networks Lab	0-1-2	2
7	CD17010	Machine Learning Lab	0-1-2	2
			Total Credits	**20**

5th Semester

S.No.	Code	Course Title	L-T-P	Credits
1	CD18001	Cryptography and Network Security	4-0-0	4
2	CD18003	Internet of Things	3-0-0	3
3	CD180XX	Professional Elective – V	3-0-0	3
4	CD180XX	Professional Elective – VI	3-0-0	3
5	CD180XX	Professional Elective – VII	3-0-0	3
6	CD18005	IoT and Network Security Lab	0-1-2	2
7	CD18091	Short Term Industrial / Research Experience*	0-0-0	2
8	CD18093	Comprehensive Viva-Voce	0-0-0	2
			Total Credits	**22**

6th Semester

S.No.	Code	Course Title	L-T-P	Credits
1	CD18090	Seminar & Technical Writing	0-0-0	2
2	CD18098	Dissertation	0-0-0	12
			Total Credits	**14**

Professional Elective Courses:

Professional Elective- I / II		
S.No.	**Code**	**Course Title**
1	CD17021	Artificial Intelligence
2	CD17023	Big Data Technologies
3	CD17025	Distributed Operating Systems
4	CD17027	Foundations of Data Science
5	CD17029	Full Stack Development
6	CD17031	Information Systems Management
7	CD17033	Modeling and Simulation
8	CD17035	Soft Computing Techniques

Professional Elective- III / IV		
S.No.	**Code**	**Course Title**
1	CD17022	Agri-Bioinformatics
2	CD17024	Design Patterns
3	CD17026	ERP and Supply Chain Management
4	CD17028	Human Computer Interaction
5	CD17030	Image Processing
6	CD17032	Mobile Computing
7	CD17034	Natural Language Processing
8	CD17036	Software Architecture

Professional Elective- V / VI / VII		
S.No.	**Code**	**Course Title**
1	CD18021	Cloud Computing
2	CD18023	Deep Learning
3	CD18025	Digital Forensics
4	CD18027	E-Commerce Technologies and Management
5	CD18029	Foundations of Blockchain
6	CD18031	Game Theory
7	CD18033	GPU Programming
8	CD18035	Information Security
9	CD18037	Malware Detection and Mitigation
10	CD18039	Network Programming
11	CD18041	Program Analysis and Verification
12	CD18043	Quantum Computing
13	CD18045	Reinforcement Learning
14	CD18047	Social Media Analytics
15	CD18049	User Interface Design

Basic Science Courses

S.No.	Code	Course Title	L-T-P	Credits
1	MA16061	Statistics and Queuing Theory	3-0-0	3

Engineering Science Courses

S.No.	Code	Course Title	L-T-P	Credits
1	CD16001	Problem Solving through Programming	4-0-0	4
2	CD16007	Programming Lab	0-1-2	2
3	CD18091	Short Term Industrial / Research Experience*	0-0-0	2
4	CD18090	Seminar & Technical Writing	0-0-0	2

Humanities and Social Science Courses

S.No.	Code	Course Title	L-T-P	Credits
1	MS16064	Management Theory and Practice	3-0-0	3

SYLLABI

Master of Computer Applications

1st Semester

MA16061 **3-0-0 (3)**

Statistics and Queuing Theory

Pre-Requisites: **None**

Course Outcomes:

CO-1	Make use of the concept of random variables to solve problems arising in probability distributions
CO-2	Explain the concepts of estimates
CO-3	Perform hypothesis tests on small and large samples
CO-4	Find the coefficient of correlation and lines of regression
CO-5	Understand the characteristics of a queueing model

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	3	–	3	1	–	2
CO-2	3	–	3	–	–	2
CO-3	3	–	3	–	–	2
CO-4	3	–	3	–	–	2
CO-5	3	–	–	2	–	–

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Random Variables and their Distributions: Review of Probability concepts and Bayes Theorem; Random variables (discrete and continuous), probability functions, density and distribution functions, special distributions (Binomial, Hypergeometric, Poisson, Uniform, exponential and normal), Mean and variance, Chebyshev's inequality, Joint probability mass function, marginal distribution function, joint density function

Testing of Estimates: Estimation, point and interval estimates, unbiased and efficient estimators, estimation of parameters by the method of moments and maximum likelihood method

Testing of Hypothesis: Testing of Hypothesis, Null and alternative hypothesis, level of significance, one-tailed and two-tailed tests, tests for large samples (tests for single mean, difference of means), tests for small samples (t, F, and Chi-square tests), goodness of fit, contingency tables, analysis of variance (one-way classification), Non-parametric tests, regression, correlation

Queueing Theory: Concepts, applicability, classification, birth and death process, Poisson queues, single server, multiple servers, queueing models, infinite (including waiting times) and finite capacities, Erlangen distribution, Erlangen service time queueing models

Learning Resources:

Text Books:

1. Miller & Freund's Probability and Statistics for Engineers, Richard A. Johnson, Pearson, 2018, Ninth Edition.
2. Modern Elementary Statistics, John E. Freund and Benjamin M. Perles, Pearson, 2013, Twelfth

Edition.
3. Operations Research: An Introduction, Hamdy A. Taha, Pearson, 2017, Tenth Edition

Reference Books:

1. Fundamentals of Mathematical Statistics, S.C. Gupta and V.K. Kapoor, Sultan Chand & Sons, 2020, Twelfth Edition.
2. Operations Research, Kanti Swarup, P.K. Gupta and Man Mohan, Sultan Chand & Sons, 2014.

CD16001 **4-0-0 (4)**

Problem Solving through Programming

Pre-Requisites: **None**

Course Outcomes:

CO-1	Design algorithms for solving simple mathematical problems including computing, searching and sorting
CO-2	Compare and contrast algorithms in terms of space and time complexity to solve simple mathematical problems
CO-3	Explore the internals of computing systems to suitably develop efficient algorithms
CO-4	Examine the suitability of data types and structures to solve specific problems
CO-5	Apply control structures to develop modular programs to solve mathematical problems
CO-6	Apply object-oriented features in developing programs to solve real-world problems

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	3	1	2	2	2	1
CO-2	2	1	2	2	2	1
CO-3	1	2	3	1	3	1
CO-4	1	2	1	2	2	1
CO-5	3	3	1	2	2	2
CO-6	2	3	1	2	3	2

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Problem Solving – Algorithm – Procedural Abstraction – Functions – Parameter Passing – Recursion– Structures – Classes – Arrays – Pointers – Inheritance – polymorphism – Overloading – Templates.

Learning Resources:

Text Books/Reference Books:

1. Walter Savitch, Problem Solving with C++, 2/e, Pearson, 2002
2. Cay Horstmann, Timothy Budd, Big C++, Wiley, Indian Edition, 2006 Website reference links

CD16003 **3-0-0 (3)**

Computer Organization

Pre-Requisites: **None**

Course Outcomes:

CO-1	Fundamentals of digital computers, and internal organization of computers
CO-2	Analysis of the CPU, memory unit, Input/Outputs and the relations between its main components.
CO-3	Analysis of the memory hierarchy and perform computations with the functional units of the processor and memory.
CO-4	Cost performance and design trade-offs in designing and constructing a computerprocessor including memory.
CO-5	Apply the knowledge of combinational and sequential logical circuits to design computer architecture to calculate the arithmetic expressions.

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	1	1	1	1	1	-
CO-2	2	1	2	2	1	1
CO-3	3	1	1	2	2	1
CO-4	3	2	1	2	2	1
CO-5	2	3	2	2	-	1

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Basic Structure – Functional Units – Bus Structure – Addressing Methods – Machine Program Sequence – 68000 example – Instructions – Assembly Language Program – Flow Control – Power PC example – Processing Unit – Hardwired Control – Micro Programmed Control – Memory – Performance Considerations – Arithmetic and Branching Conditions – Computer Peripherals

Learning Resources:

Text Books:

1. Hamacher, Vranesic, Zaky, Computer Organization, 5/e, MGH,
2. William Stallings, Compute Organization and Architecture Designing for Performance, 8/e, Pearson Education, 2010.

CD16005 **3-0-0 (3)**

Discrete Mathematics

Pre-Requisites: **None**

Course Outcomes:

CO-1	Apply formal methods of proof to solve discrete problems
CO-2	Apply Propositional logic and First order logic to solve problems
CO-3	Apply techniques for counting the occurrences of discrete events including permutations, combinations with or without repetitions
CO-4	Formulate and solve graph problems including searching and spanning
CO-5	Formulate and solve recurrence relations

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	3	2	1	2	1	1
CO-2	3	2	1	1	1	1
CO-3	3	3	-	1	1	1
CO-4	3	3	2	2	1	1
CO-5	3	2	1	1	1	1

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Sets, Relations, Functions – Fundamentals of Logic – Quantified Propositions – Mathematical Induction – Combinations and Permutations – Enumerations – Recurrence Relations – Generating Functions – Binary Relations – Lattices – Directed Graphs – Graphs – Spanning Trees – Planar Graphs – Euler Circuits – Hamiltonian Graphs

Learning Resources:

Text Books:

1. Joe L. Mott, Abraham Kandel, Theodore P. Baker, Discrete Mathematics for Computer Scientists and Mathematicians, Second Edition, PHI, 2001.
2. Tremblay J. P. and Manohar R., Discrete Mathematical Structures, MGH, 1997.
3. Kenneth H. Rosen, Discrete Mathematics and Its Applications with Combinatorics and Graph Theory, Seventh Edition, MGH, 2011.

CD16007 **0-1-2 (2)**

Programming Lab

Pre-Requisites: **None**

Course Outcomes:

CO-1	Design and test programs to solve mathematical and scientific problems
CO-2	Design and test complex control structures for real-life problems
CO-3	Implement modularity for complex, reliable and robust structures
CO-4	Design solutions for the real-life problems using OOP paradigm

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	3	2	2	1	2	1
CO-2	1	1	2	1	2	2
CO-3	1	2	3	2	2	1
CO-4	2	2	2	2	2	3

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Familiarization – Editing – Expressions – Series Evaluation – Functions – Recursion – Arrays – Pointers – Structures – Classes – Inheritance – Polymorphism – Overloading – Templates

Learning Resources:

Text Books:

1. Walter Savitch, Problem Solving with C++, Ninth Edition, Pearson, 2014.
2. Cay Horstmann, Timothy Budd, Big C++, Wiley, 2nd Edition, 2009.
3. R.G. Dromey, How to solve it by Computer, Pearson, 2008.

CD16009 **3-0-0 (3)**

Web Programming

Pre-Requisites: **None**

Course Outcomes:

CO-1	Understand, analyze and build dynamic and interactive websites
CO-2	Understand current and evolving Web languages for integrating media and user interaction in both front end and back end elements of a Web site
CO-3	Analysis and reporting of web data using web analytics
CO-4	Applying different testing and debugging techniques and analyzing the website effectiveness.

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	2	1	-	2	3	1
CO-2	-	1	-	2	3	3
CO-3	-	1	2	2	3	1
CO-4	-	1	-	2	3	2

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Web Essentials - Client-Server Architecture of Internet – File Transfer Protocol - HTTP, HTTPS - SMTP–POP3 - Client Side Programming -Creating Home Pages - Dynamic HTML - JavaScript - Cascading Style Sheets - Including Multimedia - Web Servers- Server side Programming- String Processing and Regular Expressions, Form Processing and Business Logic, Dynamic Content - Database Connectivity - Application Development.

Learning Resources:

Text Books:

1. Deitel, Deitel & Nieto, Internet and Worldwide Web - How to Program, 5th Edition, PHI, 2011.
2. Bai and Ekedhi, "The Web Warrior Guide to Web Programming", 3rd Edition, Thomson, 2008.
3. Thomas A Powel, "Web Design: The Complete Reference", 2nd Edition, McGraw-Hill, 2002

CD16011 **0-1-2 (2)**

Web Programming Lab

Pre-Requisites: **None**

Course Outcomes:

CO-1	Understand, analyze and build dynamic and interactive websites
CO-2	Understand current and evolving Web languages for integrating media and user interaction in both front end and back end elements of a Web site
CO-3	Analysis and reporting of web data using web analytics
CO-4	Applying different testing and debugging techniques and analyzing the website effectiveness.

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	2	1	-	2	3	1
CO-2	-	1	-	2	3	3
CO-3	-	1	2	2	3	1
CO-4	-	1	-	2	3	2

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Creating home pages - Online shopping - Online examination - Chat system - Mailing system.

Learning Resources:

Text Books:

1. Deitel, Deitel and Nieto, Internet and Worldwide Web - How to Program, 5th Edition, PHI, 2011.
2. Bai and Ekedhi, The Web Warrior Guide to Web Programming, 3rd Edition, Thomson, 2008.

2nd Semester

MS16064 **3-0-0 (3)**

Management Theory and Practice

Pre-Requisites: **None**

Course Outcomes:

CO-1	Apply management theories, models and principles to practice through analysis of case studies and contemporary business events
CO-2	Outline the historical evolution of management theories
CO-3	Describe the four management functions of planning, organizing, leading, and controlling
CO-4	Understand different organizational functions and business processes

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	–	2	–	–	–	3
CO-2	–	2	–	–	1	3
CO-3	–	2	–	–	1	3
CO-4	–	2	–	–	1	3

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Introduction to Management: Definition of Management, Science or Art, Manager vs. Entrepreneur, types of managers, managerial roles and skills, Evolution of Management, Organization culture and Environment, Current trends and issues in Management.

Planning: Nature and purpose of planning, planning process, types of planning, Objectives, Planning premises, Planning Tools and Techniques, Decision making steps and process.

Organising: Nature and purpose, Formal and informal organization, organization chart, organization structure, types, Line and staff authority, departmentalization, delegation of authority, centralization and decentralization, Managing Human Resources.

Directing: Foundations of individual and group behaviour, motivation, motivation theories, job satisfaction, job enrichment, leadership – types and theories of leadership, Communication, process of communication, barrier in communication, effective communication.

Controlling: System and process of controlling, budgetary and non-budgetary control techniques, Productivity problems and management, control and performance, direct and preventive control – reporting.

Learning Resources:

Text Books:

1. Harold Koontz, Heinz Weihrich, Mark V. Cannice, Essentials of Management, TMH, 2020, 11th Edition.
2. Gerald A Cole, Phil Kelly, Management Theory and Practice, Cengage, 2020, 9th Edition

Reference Books:

1. Management, Stephen P. Robbins, Mary Coulter, Agna Fernandez, Pearson, 2019, 14th Edition.
2. Management, Richard L. Daft, Cengage, 2018, 13th Edition.

CD16002 **3-0-2 (4)**

Software Engineering

Pre-Requisites: **None**

Course Outcomes:

CO-1	Comprehend software development life cycle
CO-2	Prepare SRS document for a project
CO-3	Apply software design and development techniques
CO-4	Identify verification and validation methods in a software engineering project
CO-5	Implement testing methods for software
CO-6	Analyze and Apply project management techniques for a case study

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	1	2	1	2	1	1
CO-2	1	2	2	3	2	1
CO-3	1	1	2	3	1	1
CO-4	1	1	1	3	3	1
CO-5	1	1	1	3	2	1
CO-6	1	1	1	3	3	1

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

The Software Process – Requirements, Specification, Design, Implementation and Maintenance – Cohesion – Data Encapsulation – Reusability – Software Life Cycle – Use-case modeling – Data flow Analysis – Transaction Analysis – 4GL – Coding Standards – Module reuse – Module Testing – CASE tools for integration and Complete Software Process –Project planning and Project scheduling- Software Metrics –Cost estimation using models like COCOMO – Risk management activities–Maintenance Models– Configuration Management –Reverse Engineering–Software Re-engineering.

Learning Resources:

Text Books:

1. Stephen R Schach, Object Oriented and Classical Software Engineering, 5/e, TMH, 2010
2. Ian Sommerville, Software Engineering, 9/e, Pearson, 2010.

CD16004 **3-0-0 (3)**

Data Structures

Pre-Requisites: **None**

Course Outcomes:

CO-1	Analysis of algorithms and fundamental concepts of ADT, identification of suitable data structures to solve complex problems
CO-2	Fundamentals of Arrays, Stack and Queues, develop and analysis of the algorithms fordifferent operations.
CO-3	Fundamentals of Binary Trees and Graphs, develop algorithms for various operations performed on trees and graphs.
CO-4	Implementation and analysis of sorting and searching techniques
CO-5	Implement symbol table using hashing techniques and multi-way search trees

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	1	2	-	1	1	2
CO-2	2	1	1	2	2	1
CO-3	2	3	1	1	2	2
CO-4	2	2	2	3	2	2
CO-5	2	3	1	2	1	1

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Order Lists- Stacks- Queues-Trees-Search Trees-BST, AVL Tree - Hashing - Hash Tables - Priority Queues - Internal and External Sorting - Disjoint Sets - Graph Algorithms - Shortest Paths - Spanning Trees.

Learning Resources:

Text Books:

1. Mark Allen Weiss, Data Structures and Algorithm Analysis in C++, 2/e, Pearson, 2004.
2. Sartaj Sahni, Data Structures, Algorithms and Applications in C++, 2/e, University Press, 2005

CD16006 **3-0-0 (3)**

Object Oriented Programming

Pre-Requisites: **None**

Course Outcomes:

CO-1	Understand fundamental concepts in an object-oriented approach.
CO-2	Analyze design issues in developing OOP applications.
CO-3	Write computer programs to solve real-world problems in Java.
CO-4	Analyze source code API documentations.
CO-5	Create GUI-based applications.

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	1	1	-	1	1	-
CO-2	1	3	-	2	3	2
CO-3	2	2	2	2	2	2
CO-4	1	2	-	3	2	3
CO-5	2	2	1	2	2	2

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Object Oriented Thinking – Messages and Methods – OO Design – Software Components – Design Paradigms – Inheritance – Mechanisms for software reuse – Polymorphism – AWT Classes – Input output Streams – Design Patterns – Exception handling

Learning Resources:

Text Books:

1. Timothy Budd, "Understanding Object Oriented Programming with Java", Pearson Education, 1999
2. Herbert Schildt, "Java 2 Complete Reference", 5/e, TMH, 2010

CD16008 **3-0-0 (3)**

Algorithm Design

Pre-Requisites: **None**

Course Outcomes:

CO-1	Analyse time and space complexity
CO-2	Identify algorithm design methodology to solve problems
CO-3	Design algorithms for network flows
CO-4	Distinguish between P and NP classes of problems
CO-5	Analyse amortized time complexity

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	2	2	-	1	3	1
CO-2	2	2	1	2	2	1
CO-3	2	3	-	2	2	1
CO-4	2	1	-	-	3	1
CO-5	2	2	-	1	3	1

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Algorithm Analysis – Asymptotic notation– Greedy method – Divide and conquer – Dynamic programming – example problems in each case of design methods – Sorting Algorithms – Graph Algorithms – Shortest path, search algorithms, Minimum spanning tree – Strings and Pattern matching Algorithms – Backtracking, and Branch and Bound methods - P, NP, NP-hard, NP-complete classes.

Learning Resources:

Text Books:

1. M T Goodrich, Roberto Tamassia, Algorithm Design, John Wiley & Sons, 2001
2. Horowitz, Sartaj Sahni, S Rajasekaran, Computer Algorithms, 2/e, Silicon Pr., 2007

CD16010 **0-1-2 (2)**

Data Structures Lab

Pre-Requisites: **None**

Course Outcomes:

CO-1	Develop ADT for stack and queue applications
CO-2	Implement tree and graph algorithms
CO-3	Implement and analyze internal and external sorting algorithms
CO-4	Design and implement symbol table using hashing technique

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	2	1	-	2	2	1
CO-2	2	1	1	2	2	1
CO-3	3	2	1	3	1	1
CO-4	3	2	2	2	2	2

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Implementation of ordered lists – Generic Queues – conversion of expressions, evaluation, expression trees – Search Trees – BST – AVL Trees – Splaying – Sorting algorithms – Graph traversals – Shortest paths – Spanning Trees

Learning Resources:

Text Books:

1. Mark Allen Weiss, Data Structures and Algorithm Analysis in C++, 2/e, Pearson, 2004.
2. Sartaj Sahni, Data Structures, Algorithms and Applications in C++, 2/e, University Press, 2005

CD16012 **0-1-2 (2)**

Object Oriented Programming Lab

Pre-Requisites: **None**

Course Outcomes:

CO-1	Understand fundamental concepts in an object-oriented approach.
CO-2	Analyze design issues in developing OOP applications.
CO-3	Write computer programs to solve real-world problems in Java.
CO-4	Analyze source code API documentations.
CO-5	Create GUI-based applications.

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	1	1	-	1	1	-
CO-2	1	3	-	2	3	2
CO-3	2	2	2	2	2	2
CO-4	1	2	-	3	2	3
CO-5	2	2	1	2	2	2

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Classes, declaring objects in classes, Methods, constructors, garbage collection, Method overloading, passing objects as parameters, Inheritance, various forms and types of inheritance, Multilevel hierarchy, use of super, method overriding, Applications of method overriding, abstract classes, Packages with examples

Interfaces and implementation, Exception handling, types, throwing, creating own exceptions, Multithreading and concepts, its usage and examples, Input/output streams, String operations and examples, Collection classes-array, stack collection, bitset collection, Utility classes-string tokenizer, bitset, date, Applets- methods, creation, designing and examples, Event handling- event classes, Event listener interfaces, AWT classes, working with frames, AWT controls- layout manager, user interface components, Graphics programming

Learning Resources:

Text Books:

1. Timothy Budd, Understanding object-oriented programming with Java, Pearson, 2000.
2. Herbert Schildt, The complete reference Java 2, TMH, 2017.

3rd Semester

CD17001 **3-0-2 (4)**

Python Programming

Pre-Requisites: **None**

Course Outcomes:

CO-1	Understand the fundamental concepts of Python programming
CO-2	Develop programs based on iterations and control statements
CO-3	Analyse the various data formats and their representations
CO-4	Use various Scientific computing tools and databases
CO-5	Generate the various data visualizations

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	2	2	1	2	1	1
CO-2	2	2	1	2	3	1
CO-3	2	2	2	2	2	2
CO-4	1	3	2	2	2	1
CO-5	1	1	1	1	1	3

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Introduction to Python Programming Language, Features of Python Programming Language , Flavours of Python Programming Language , Installation of Python Programming Language , Execution of First Python Program, Data Types , Fundamental Data Types , Collection Data Types – Lists, Tuples, Sets, Frozen sets, Dictionaries –, Variables , Operators, Control Statements , Conditional Statements, Looping Statements , Logical Programs, String Handling, Functions, File handling, Object Oriented Programming in Python, Exception Handling in Python, Class and Objects, Different modules and packages for scientific computing in python (Numpy, Pandas, Matplotlib, Scipy, Sympy, etc.) , Database access, Regular Expressions.

Learning Resources:

Text Books:

1. Python The Complete Reference, Martin C Brown
2. Learning with Python: How to Think Like a Computer Scientist, Allen Downey, Jeff Elkner, and Chris Meyers.
3. https://www.w3schools.com/python/
4. https://www.javatpoint.com/python-tutorial

CD17003 **3-0-0 (3)**

Operating Systems Principles

Pre-Requisites: **CD16003**

Course Outcomes:

CO-1	Distinguish functional architectures of operating systems and file systems
CO-2	Develop algorithms for subsystem components
CO-3	Design device drivers and multi-threading libraries for an OS
CO-4	Develop application programs using UNIX system calls
CO-5	Design and solve synchronization problems

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	1	1	-	1	1	1
CO-2	2	2	-	2	2	1
CO-3	2	2	-	1	2	2
CO-4	2	2	-	2	2	1
CO-5	2	3	-	2	1	1

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Batch, iterative, time-sharing and real-time systems – Operating system structure – Concurrent processes – Synchronization – CPU scheduling – Deadlocks – Memory management – Virtual memory -Secondary storage management – File systems – I/O systems – Mass-storage structure – Protection – Security.

Learning Resources:

Text Books:

1. A. Silberschatz, Galvin, Gagne, Operating System Concepts, 8/e, John Wiley & Sons, 2009.
2. Andrew S Tanenbaum, Modern Operating Systems, 3/e, Pearson Education, 2007

CD17005 **3-0-0 (3)**

Principles of Database Management Systems

Pre-Requisites: **None**

Course Outcomes:

CO-1	Understand functional components of the DBMS.
CO-2	Devise queries using Relational Algebra, Relational Calculus and SQL.
CO-3	Design database schema.
CO-4	Develop E-R model.
CO-5	Evaluate and optimize queries.
CO-6	Analyze transaction processing, concurrency control and recovery techniques.

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	2	-	1	-	1	1
CO-2	2	2	2	2	2	1
CO-3	2	3	2	2	3	1
CO-4	2	2	1	1	1	2
CO-5	2	1	1	1	3	1
CO-6	2	1	1	1	2	1

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Introduction to DBMS: Historical perspective, File Versus a DBMS, Advantages of DBMS, Describing and storing data in DBMS, Architecture of a DBMS, Different Data Models;

Entity Relationship(ER) model: Features of ER model, conceptual design using ER model, design for large enterprises; Relational model– structure and operations, Integrity constraints over relations;

Query languages: Relational Algebra, Relational Calculus and SQL– Queries, Constraints, Form of SQL query, UNION, INTERSECT and EXCEPT, Nested queries, Aggregate Operators, Null values, Complex Integrity constraints in SQL, triggers and Embedded SQL;

Database Design: Mapping ER model to Relational form; Functional Dependency–Closer of functional dependencies, closer of attributes, canonical cover and Properties of Decompositions; Normalization process – 1NF, 2NF, 3NF and BCNF; Multivalued dependency– Closer properties of Multivalued dependency and 4NF; Join dependency– PJNF, Decomposition Algorithms;

Transaction Management: ACID properties, transactions, schedules and concurrent execution of transactions; Concurrency control – lock based protocol, Serializability, recoverability, dealing with deadlocks and Concurrency control without locking;

Query Processing: Overview of Query Evaluation, operator evaluation; Algorithms for relational operations– Selection operation, General selection condition, Projection operation, Join operation, set operation and aggregate operation, Evaluation of relational operations; Query optimization: Alternative

plans, functions of query optimizer, translating SQL queries into relational algebra, estimating the cost of a plan, relational algebra equivalences, and other approaches to query optimization;

Database Recovery: Failure classification, Recovery and atomicity, Log-based recovery shadow paging and Advanced Recovery Techniques:

Security and Authorization: Access control, direct access control and Mandatory access control, Role of DBA, Application development.

Learning Resources:

Text Books:

1. Elamsri, Navathe, Somayajulu and Gupta, Fundamentals of Database Systems, 6thEdition, Pearson Education, 2011.
2. Raghu Ramakrishnan, Johannes Gehrke, Database Management Systems, 3nd Edition, McGraw Hill, 2003.
3. Silberschatz, Korth and Sudharshan, Database System Concepts, 6rd Edition, McGrawHill, 2010.

CD17007 **0-1-2 (2)**

Database Management Systems Lab

Pre-Requisites: **None**

Course Outcomes:

CO-1	Design and Implement a database schema
CO-2	Devise queries using DDL, DML, DCL and TCL commands.
CO-3	Develop application programs using PL/SQL
CO-4	Design and implement a project using embedded SQL and GUI.
CO-5	Apply modified components for performance tuning in open source software.

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	2	2	2	2	2	2
CO-2	2	1	2	2	2	2
CO-3	2	2	2	2	2	2
CO-4	2	2	2	2	2	1
CO-5	2	1	1	2	3	1

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Familiarization of Oracle RDBMS, SQL*Plus and Oracle developer,

SQL: query-structure; DDL-create, alter, drop, rename and Truncate; DML-select, insert, update, delete and lock; Set operations- union, intersection and except; join; Aggregate Operations- group-by and having; nested sub-queries and views; DCL-grant and revoke, TCL- Commit, save point, rollback and set transaction.

PL/SQL: Environment, block structure, variables, operators, data types, control structures; Cursors structures- Implicit and Explicit; Bulk statements- Bulk collect into and forall; Exception handling- Compilation and Run-time, user-defined; Stored procedures- creation options, pass- by-value and functions-pass-by-value; Packages-package specification, body, package creation and usage; Triggers- Data definition language triggers, Data manipulation triggers, Compound triggers and trigger restrictions; Large objects-CLOB, NCLOB, BLOB and BFILE; Implementation of applications using GUI; group project;

Learning Resources:

Text Books:

1. James, Paul and Weinberg, Andy Oppel, SQL: The Complete Reference, 3rd Edition, McGraw Hill, 2011.
2. Michael McLaughlin, Oracle Database 11g PL/SQL Programming, Oracle press.

CD17009 **0-1-2 (2)**

Operating Systems Lab

Pre-Requisites: **None**

Course Outcomes:

CO-1	Implement elementary UNIX system commands
CO-2	Develop programs to test synchronization problems
CO-3	Design and develop user level thread library
CO-4	Design and implement file system

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	1	1	-	1	2	1
CO-2	2	3	-	2	1	1
CO-3	2	2	-	2	2	1
CO-4	2	2	-	1	2	2

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Study of race conditions – Use of semaphores to solve concurrency problems – Implementation of critical region construct, monitor – Comparison of different scheduling algorithms – Implementation of Dekker's algorithm – Implementation of memory manage

Learning Resources:

Text Books:

1. A. Silberschatz, Galvin, Gagne, Operating System Concepts, 8/e, John Wiley & Sons, 2009
2. Andrew S Tanenbaum, Modern Operating Systems, 3/e, Pearson Education, 2007.

4th Semester

CD17002 **3-0-2 (4)**

Software Testing

Pre-Requisites: **None**

Course Outcomes:

CO-1	Analyze Various test processes and continuous quality improvement
CO-2	Analyze Types of errors and fault models
CO-3	Modeling the behavior using FSM
CO-4	Application of software testing techniques in commercial environments

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	2	2	3	2	2	–
CO-2	3	2	3	1	2	–
CO-3	3	2	3	2	2	–
CO-4	3	2	2	2	2	–

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Introduction, Flow graphs and Path testing, Transaction Flow Testing, Dataflow testing, Domain Testing, Paths, Path products and Regular expressions, Logic Based Testing, Specifications, State, State Graphs and Transition testing, Graph Matrices and Application

Learning Resources:

Text Books:

1. Baris Beizer, Software Testing Techniques,2/e, Dreamtech,1990
2. Perry, "Effective methods of Software Testing", John Wiley.,2006

CD17004 **3-0-0 (3)**

Computer Networks

Pre-Requisites: **None**

Course Outcomes:

CO-1	Understand OSI and TCP/IP models
CO-2	Analyze MAC layer protocols and LAN technologies
CO-3	Design applications using internet protocols
CO-4	Implement routing and congestion control algorithms
CO-5	Develop application layer protocols

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	-	1	-	-	-	-
CO-2	-	1	1	-	-	-
CO-3	3	3	-	3	-	-
CO-4	1	2	1	-	-	-
CO-5	3	3	-	3	-	-

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Network structures – Network Architecture – OSI model – LAN protocols – IEEE standard 802 – Ethernet – Token Bus and Token Ring – Error Detection and Correction – Sliding Window protocols

Routing algorithms – Congestion control algorithms – Internetworking Network Layer in Internet IP – Transport Layer in Internet – UDP, TCP – Remote Procedure Call – Implementation and semantics of RPC – E-mail Protocol and File Transfer Protocol.

Learning Resources:

Text Books:

1. A. S Tanenbaum, Computer Networks, 4/e, PHI, 2003.
2. Larry L Peterson, Bruce S Davis, Computer Networks, 5/e,Elsevier, 2012.

CD17006 **3-0-0 (3)**

Machine Learning

Pre-Requisites: **CD17001 and MA16061**

Course Outcomes:

CO-1	Understand instance-based learning and Unsupervised algorithms
CO-2	Design Neural network to solve classification and function approximation problems
CO-3	Design Regression model to solve the classification and prediction problems
CO-4	Build Optimal classifiers using Decision Trees, Naïve Bayesian and Support vector machines

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	2	1	1	2	1	1
CO-2	2	3	2	2	1	1
CO-3	2	3	2	2	1	1
CO-4	2	3	3	2	2	1

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Introduction – Well defined learning problems, Designing a Learning System, Issues in Machine Learning; - The Concept Learning Task – Feature Representations-Local and Global; Decision Tree Learning - Decision tree learning algorithm. Inductive bias- Issues in Decision tree learning; - Artificial Neural Networks – Perceptrons, Gradient descent, Multilayer networks, Backpropagation Algorithm- Convergence, Generalization; – Evaluating Hypotheses – Estimating Hypotheses Accuracy, Basics of sampling Theory, Comparing Learning Algorithms; - Bayesian Learning – Bayes theorem, Concept learning, Bayes Optimal Classifier, Naïve Bayes classifier, Bayesian belief networks, Clustering Methods, EM algorithm, GMM; - Computational Learning Theory – Sample Complexity for Finite Hypothesis spaces, Sample Complexity for Infinite Hypothesis spaces, The Mistake Bound Model of Learning; - Instance-Based Learning – k-Nearest Neighbour Learning, Locally Weighted Regression, Case-based learning - Models of Evolution and Learning; Ensemble Methods- Bagging and Boosting; SVD Dimensionality Reduction-PCA, LDA; Reinforcement Learning - The Learning Task, Q Learning, Support vector Machines, Kernel Methods an illustrative example.

Learning Resources:

Text Books:

1. Tom.M.Mitchell, Machine Learning, McGraw Hill International Edition, 1997.
2. C Bishop – Pattern Recognition and Machine Learning – Springer, 2006.
3. Duda, Richard, Peter Hart, and David Stork. Pattern Classification. 2nd ed. New York, NY: Wiley-Interscience, 2000.

CD17008 **0-1-2 (2)**

Computer Networks Lab

Pre-Requisites: **None**

Course Outcomes:

CO-1	Develop programs for client-server applications
CO-2	Perform packet sniffing and analyze packets in network traffic.
CO-3	Implement error detecting and correcting codes
CO-4	
CO-5	

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	3	3	-	2	-	-
CO-2	2	2	2	1	1	-
CO-3	1	1	2	-	-	-

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Error Correction and Detection – IP address Conversion functions – Client Server example using Pipes, FIFOs, Message Queues, Shared Memory – Connection Oriented Client Server with TCP – Connectionless Client Server with UDP – Concurrent Server – Multi- protocol Server – Internet Super Server – Chat Server – Mail Server.

Learning Resources:

Text Books:

1. A. S Tanenbaum, Computer Networks, 4/e, PHI, 2003.
2. Larry L Peterson, Bruce S Davis, Computer Networks, 5/e,Elsevier, 2012.

CD17010 **0-1-2 (2)**

Machine Learning Lab

Pre-Requisites: **CD17001**

Course Outcomes:

CO-1	Implement instance-based learning and Unsupervised algorithms
CO-2	Design Neural network to solve classification and function approximation problems
CO-3	Design Regression model to solve the classification and prediction problems
CO-4	Build Optimal classifiers using Decision Trees, Naïve Bayesian and Support vector machines

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	2	1	1	2	1	1
CO-2	2	3	2	2	1	1
CO-3	2	3	2	2	1	1
CO-4	2	3	3	2	2	1

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

1. Design and implement machine learning algorithm using least means square learning rule to play checkers game. The training experience should be generated by the system playing game with itself.

2. Implement a machine learning program to play 5×5 Ticktacktoe game.

3. Design and implement a feedforward neural network with 5 inputs, 3 hidden and 1 output units. It should use a backpropagation algorithm with batch update to train the neural network to generate odd parity bit on its output given any 5-bit binary pattern on its inputs.

4. Build a decision tree model for a classification problem.

5. Implement perceptron learning algorithm and attempt to solve two input i) AND gate ii) OR gate iii) EXOR gate problems.

6. Implement the Naïve Bayesian and SVM algorithm to obtain the classifier for the given dataset.

7. Design and implement PCA and regression model for any given dataset.

8. Implement the Clustering methods for any given dataset.

Learning Resources:

Text Books:

1. Tom.M.Mitchell, Machine Learning, McGraw Hill International Edition, 1997.
2. C Bishop – Pattern Recognition and Machine Learning – Springer, 2006.
3. Duda, Richard, Peter Hart, and David Stork. Pattern Classification. 2nd ed. New York, NY: Wiley-Interscience, 2000.

5th Semester

CD18001 **4-0-0 (4)**

Cryptography and Network Security

Pre-Requisites: **None**

Course Outcomes:

CO-1	Analyze encryption algorithms.
CO-2	Apply cryptographic algorithms to build secure protocols
CO-3	Identify system vulnerabilities of communication protocols
CO-4	Design of secure protocols to solve real world scenario

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	1	-	1	-	2	-
CO-2	2	2	-	3	3	1
CO-3	-	-	3	1	1	2
CO-4	3	1	2	3	3	3

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Security Goals and Services: Definitions – Threat; Vulnearbilitis; Attacks – Classifications of attacks; Security services; Security mechanisms

Number theory: Introduction to number theory – Modular Arithmetic; Finite fields; Number theory properties – Primality testing; Fetmat's and Euler's theorem; Chinese remainder theorem; Integer factorization; discrete logarithm

Cryptographic algorithms : Private key algorithms – Classical Encryption techniques; Stream ciphers

;Block cipher modes; DES;AES; Random number generators; Public key algorithms- Principles of Public key Cryptography; RSA; Deffie-Hellman; ElGamal; Elliptic Curve Cryptography

Security mechanisms: Key management and Distribution-Certificate authorities; PKI; MAC; Hashing; Digital Signatures-Authentication protocols; Digital Signature Standard;

Introduction to network security: Network security threats; Vulnerabilities - Denial-of- service/Distributed denial-of-service attacks; Spoofing, Man-in-the-middle, Replay, TCP/Hijacking, Fragmentation attacks, Weak keys, Mathematical attacks, Social engineering, Port scanning, Dumpster diving, Birthday attacks, Password guessing, Software exploitation, Inappropriate system use, Eavesdropping, War driving, TCP sequence number attacks, War dialing/demon dialing attacks.

Internet Protocols: TCS; DNS and routing

Network Defense Tools: Firewalls- Firewall Properties; Design of firewalls; VPN's; Filtering; Intrusion detection

Security protocols – Network and transport layer security- SSL/TLS, IPsec IKE; IPsec AH, ESP; Application security- Kerberos; S/MIME; PGP; PKI;

Learning Resources:

Text Books:

1. Charlie Kaufman, Radia Perlman, Mike Speciner, Network Security: Private Communication in a Public World, Prentice Hall, 2002
2. William Stallings, Cryptography and Network Security, 6th edition Pearson Education, 2014
3. A. Menezes, P. Van Oorschot, S. Vanstone, Handbook of Applied Cryptography, CRCPress, 2004.

CD18003 **3-0-0 (3)**

Internet of Things

Pre-Requisites: **CD16004, CD16008 and CD17004**

Course Outcomes:

CO-1	Analyze the protocol Stack for the Internet of Things to address the heterogeneity in devices and networks
CO-2	Develop smart IoT Applications using smart sensor devices and cloud systems
CO-3	Development of smart mobile apps for societal applications
CO-4	Design secure protocols for IoT systems

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	2	2	2	3	-	-
CO-2	3	3	3	3	3	2
CO-3	3	3	3	3	3	2
CO-4	2	3	2	2	2	-

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Introduction to IoT- Concepts, Services, Characteristics, Challenges and Applications of IoT- Smart City, Health-Care, Architecture of IoT - ITU, IWF, Integration of OT and IT technologies, IoT Data Flow,

IoT Protocols- Protocols Architecture of IoT**,** IoT Protocol Standards, Categorization of IoT protocols, Non-IP Network Technologies of IoT, IP Network Technologies, Service Discovery Protocols for IoT, Application Protocols, Protocol Stack of Wireless IoT. **IoT System Design**- Components of IoT System, Communications Models of IoT, IoT Platforms- Open-Source & Proprietary Platforms

Sensor and Identification Technologies of IoT- Edge Devices of IoT- Sensors and Actuators, WSN, RFID, Integration of RFID and WSN Network Technologies, Connectivity of IoT-Wireless IoT- Infrastructure WN and Ad hoc Networks, MAC Protocols of WN, Non-IP Wireless Connectivity- 802.11ah, 802.15 technologies, Wireless Embedded Internet, and IoT Routing, IP Connectivity- Mobile IP, BLE over IPv6, ZigBee over IPv6, RFID over IPv6.

Service Technologies of IoT- Edge and Fog Computing in IoT, Cloud Computing in IoT. **IoT Security**-Security Requirements, Challenges in IoT Security, Vulnerabilities and Threat Analysis, Layered Attacker model, Mirai Botnet, DDoS Attacks, Secure IoT Architecture, Securing the IoT Network, and Blockchain in IoT Security

Learning Resources:

Text Books:

1. Olivier Hersent, “The Internet of Things Key Applications and Protocols”, Wiley, 2012
2. Sudip Misra, “Introduction to IoT”, Cambridge University Press; First edition , 2021
3. David Hanes, “IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things”, Cisco Press; 1st edition 2017

Reference Books:

1. Arshdeeep Bahga, Vijay Madisetti, "Internet of Things: A Hands-on Approach", Universities Press, 2015
2. Raj Kamal, "Internet of Things: Architecture and Design Principles", McGraw Hill Education private limited, 2017
3. Kai Hwang, Min Chen, "Big Data Analytics for Cloud, IoT and Cognitive Computing", Wiley, 2018.

CD18005 **0-1-2 (2)**

IoT and Network Security Lab

Pre-Requisites: **CD16004 and CD17004**

Course Outcomes:

CO-1	Analyse the protocol Stack for Internet of Things to address the heterogeneity in devices and networks
CO-2	Usage of real IoT protocols for communication
CO-3	Design IoT network that transfer the data to the end user using application protocols
CO-4	Design an IoT network to work with a Cloud Computing infrastructure

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	2	2	2	3	-	-
CO-2	3	3	3	3	3	2
CO-3	3	3	3	3	3	2
CO-4	2	3	2	2	2	-

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

IoT Lab:

1. Design a network to perform end-to-end data transfer of IPv6 Packets over IEEE 802.15.4 Networks? Create a network topology of 10 Nodes and test to check the connection between two different UDP clients and verify the correctness of the received packets. (Hint: Use 6LowPAN Module in NS3 and Validation using Wireshark, Use NetAnim for Visualization).
2. Design a network to perform end-to-end data transfer of IPv6 Packets over Bluetooth Low Energy Networks. Create a network topology of 10 Nodes and test to check the connection between different two UDP clients and verify the correctness of the received packets. (Hint: Use the 6LowPAN Module in NS3 and Validation using Wireshark; Use NetAnim for Visualization.)
3. Design an IoT Network that consists of three RPL Border routers. Each RPL Border Router has two different edge networks, and each RPL Border Router connects a regular IP network with an RPL 6LoWPAN edge network. To perform this activity create a network topology of 50 Nodes and establish the connection between them. Display and visualise the data (Hint: Contiki OS, Cooja Emulator, and Wireshark and Use Web-Sense Motes and Sky Motes).
4. Design an MQTT-SN that exchanges messages between the node and the network broker. Create a simulation with an RPL Border Router device to connect a regular IP network with an MQTT-SN network. To perform this activity, create a network topology of 50 Nodes and establish the connection between them. Display the data and visualise the data (Hint: Contiki OS, Cooja Emulator, and Wireshark).
5. Design a Wireless Sensor network that generates data, transmits it through the Internet, and stores it in an IoT Cloud. To perform this activity, create a network topology of 10 Nodes and establish a connection between them. Display the data and visualise the data (Hint: Contiki OS, Cooja Emulator, and Ubidots).

6. Design an IoT Network that consists of three RPL Border routers. Each RPL Border Router has two different edge networks, and each RPL Border Router connects a regular IP network with an RPL 6LoWPAN edge network. Store the generated data from the edge devices in the IoT Cloud. To perform this activity, create a network topology of 10 Nodes and establish the connection between them; display the data, and visualize the data. You can pause the simulation and examine the packets and console output at your own pace; simply click the Pause button at the Simulation Control panel. When you are done, click Restart. (Hint: Contiki OS, Cooja Emulator, and Ubidots; Use Web-Sense Motes and Sky Motes).

Network Security Lab:

1. Write a program to mount a DoS attack using SYN flooding?
2. Write a program to Mount DNS Cache Poisoning Attack?
3. Write a program to mount the Buffer Overflow to Spawn a Shell? Apply Defences measure
4. Write a program to Working Worm- the AbraWorm
5. Write a program to mount SQL Injection Attack, The Slowloris Attack on Web Servers , and Protection of Web Server

Learning Resources:

Text Books:

1. Olivier Hersent, “The Internet of Things Key Applications and Protocols”, Wiley, 2012
2. Sudip Misra, “Introduction to IoT”, Cambridge University Press; First edition , 2021
3. David Hanes, “IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things”, Cisco Press; 1st edition 2017

Reference Books:

1. Arshdeeep Bahga, Vijay Madisetti, "Internet of Things: A Hands-on Approach", Universities Press, 2015
2. Raj Kamal, "Internet of Things: Architecture and Design Principles", McGraw Hill Education private limited, 2017
3. Kai Hwang, Min Chen, "Big Data Analytics for Cloud, IoT and Cognitive Computing", Wiley, 2018.

CD18091 **0-0-0 (2)**

Short Term Industrial / Research Experience*

Pre-Requisites: **None**

Course Outcomes:

CO-1	Apply engineering principles to real-world problems, gaining practical experience
CO-2	Plan, manage and execute the work with ethical consideration
CO-3	Review the social and environmental impact of the work
CO-4	Communicate the learnings through report and presentation

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	3	3	3	3	3	
CO-2	2	2	2	2	2	
CO-3						3
CO-4						
CO-5	3	3	3	3	3	
CO-6	2	2	2	2	2	

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Description:

Every student has to undergo either a Summer Internship / EPICS / Research project. The summer internship may be undergone in an Industry/Research organization or any premier academic Institution, including NIT Warangal for 6-8 weeks. The EPICS/research project shall be registered under the guidance of any faculty member in the institute. The student is required to submit a report and present the work before an evaluation committee, nominated by the Head of the Department.

Evaluation Criteria:

The student will be evaluated by the panel based on the below criteria. Weightage for each criterion will be determined by the panel and will be informed to the students.

Criteria	Description	Weightages
I	Relevance of the area of work	20 Marks
II	Performance of the Task	20 Marks
III	Crucial learnings from the work	20 Marks
IV	Report Preparation	20 Marks
V	Presentation and Response to questions	20 Marks

CD18093 **0-0-0 (2)**

Comprehensive Viva-Voce

Pre-Requisites: **None**

Course Outcomes:

CO-1	Comprehend and correlate the understanding of all courses in post graduate curriculum of Computer Science and Engineering

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	3	3	3	3	3	3

1 - Slightly; **2 - Moderately;** **3 - Substantially**

6th Semester

CD18090 **0-0-0 (2)**

Seminar & Technical Writing

Pre-Requisites: **None**

Course Outcomes:

CO-1	Analyze the selected topic, organize the content and communication to audience in an effective manner
CO-2	Practice the learning by self-study

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	2	3	1	1	–	–
CO-2	2	3	1	1	–	–

1 - Slightly; **2 - Moderately;** **3 - Substantially**

CD18098 **0-0-0 (12)**

Dissertation

Pre-Requisites: **None**

Course Outcomes:

CO-1	Identify a domain specific and contemporary topic
CO-2	Review literature to identify gaps and define objectives & scope of the work.
CO-3	Develop a prototype/model, experimental set-up or software systems to meet the objectives
CO-4	Analyze the results to draw valid conclusions

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	3	2			2	1
CO-2	2	2			1	2
CO-3	2	2	3	3	3	2
CO-4	2	2		3	3	
CO-5	3	2			2	1
CO-6	2	2			1	2

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Description:

Students are expected to choose real-world contemporary problem and apply the engineering principles learned, to solve the problem through building prototypes or simulations or writing codes or establishing processes/synthesis/correlations etc. The department constituted panel will decide the suitability and worthiness of the project

The method of evaluation should be as per the guidelines stipulated for the MCA Project evaluation. The template for preparation of the project report may be downloaded from (https://www.nitw.ac.in/path/?dept=/nitwForms) under UG Forms. The students are required to submit a report showing that plagiarism is within 30%. The MCA project work will be evaluated for 100 marks, with the following weightages:

Component	**Weightage**
Periodic evaluation by Guide	40 marks
Mid-term review	20 marks
End Semester viva-voce examination	40 marks
Total	100 marks

The midterm review and the end semester viva-voce examination will be conducted by a committee constituted by the Head of the Department. If the performance of a student is not satisfactory, he/ she can be awarded ‘F’ grade. Such a student will be given a maximum time of three months to improve his/her performance. If the performance of such a student is not satisfactory even after the extended time period, he/ she will have to repeat the project work in the next academic year.

Evaluation Criteria:

The student will be evaluated by the panel based on the below criteria. Weightage for each criterion will be determined by the panel and will be informed to the students.

Criteria	Description	Weightages
I	Selection of Topic	10 Marks
II	Literature Survey	10 Marks
III	Objectives and Solution Methodology	20 Marks
IV	Performance of the Task and clarity on the work	20 Marks
V	Report Preparation	20 Marks
VI	Presentation and Response to questions	20 Marks

Professional Electives

CD17021 **3-0-0 (3)**

Artificial Intelligence

Pre-Requisites: **None**

Course Outcomes:

CO-1	Solve searching problems using A*, Mini-Max algorithms.
CO-2	Create logical agents to do inference using first order logic.
CO-3	Understand Bayesian Networks to do probabilistic reasoning.
CO-4	Perform Statistical learning using EM algorithm.

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	3	3	3	3	3	2
CO-2	3	3	2	3	3	2
CO-3	3	3	2	2	2	2
CO-4	3	3	2	2	2	1

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Introduction: Agents and Objects – Evaluation of Agents – Agent Design Philosophies - Multi-agent System – Mobile Agents – Agent Communication – Knowledge query and Manipulation Language – Case Study. What is AI?, The Foundations of Artificial Intelligence; - INTELLIGENT AGENTS – Agents and Environments, Good Behavior: The Concept of Rationality, The Nature of Environments, The Structure of Agents; - SOLVING PROBLEMS BY SEARCH – Problem-Solving Agents, Formulating problems, Searching for Solutions, Uninformed Search Strategies, Breadth-first search, Depth-first search, Searching with Partial Information, Informed (Heuristic) Search Strategies, Greedy best-first search, A* Search: Minimizing the total estimated solution cost, Heuristic Functions, Local Search Algorithms and Optimization Problems, Online Search Agents and Unknown Environments; –ADVERSARIAL SEARCH – Games, The minimax algorithm, Optimal decisions in multiplayer games, Alpha-Beta Pruning, Evaluation functions, Cutting off search, Games that Include an Element of Chance; - LOGICAL AGENTS – Knowledge-Based agents, The Wumpus World, Logic, Propositional Logic: A Very Simple Logic, Reasoning Patterns in Propositional Logic, Resolution, Forward and Backward chaining; -

First Order Logic: – Syntax and Semantics of First-Order Logic, Using First-Order Logic, Knowledge Engineering in First-Order Logic;

Inference in First Order Logic: – Propositional vs. First- Order Inference, Unification and Lifting, Forward Chaining, Backward Chaining, Resolution; - UNCERTAINTY – Acting under Uncertainty, Basic Probability Notation, The Axioms of Probability, Inference Using Full Joint Distributions, Independence, Bayes' Rule and its Use, The Wumpus World Revisited; - PROBABILISTIC REASONING – Representing Knowledge in an Uncertain Domain, The Semantics of Bayesian Networks, Efficient Representation of Conditional Distribution, Exact Inference in Bayesian Networks, Approximate Inference in Bayesian Networks; - STATISTICAL LEARNING METHODS – Statistical Learning, Learning with Complete Data, Learning with Hidden Variables: EM Algorithm.

Learning Resources:

Text Books:

1. Stuart Russell, Peter Norvig, Artificial Intelligence -A ModernApproach, 3/e, Pearson, 2003.
2. Nils J Nilsson, Artificial Intelligence: A New Synthesis, MorganKaufmann Publications, 2000.

CD17023 **3-0-0 (3)**

Big Data Technologies

Pre-Requisites: **None**

Course Outcomes:

CO-1	Understand big data challenges in different domains including social media, transportation, finance and medicine
CO-2	Analyze the capability of No-SQL systems
CO-3	Apply machine learning algorithms for data analytics
CO-4	Analyze MAP-REDUCE programming model for better optimization
CO-5	Analyze the capability of Stream Data Processing Systems

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	1	1	3	1	1	-
CO-2	1	2	3	2	1	-
CO-3	1	2	3	3	1	-
CO-4	1	3	3	2	2	-
CO-5	-	1	1	1	2	-

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Overview of Big Data Analytics: Big Data Characteristics, Big Data Challenges, Big Data Analytics system Architecture, Big Data Storage Technologies, Big Data Analytics Life Cycle; No SQL databases: Introduction to No SQL databases, Types of No SQL databases, CAP Theorem, Replication & Shading, MongoDB basics; Overview of Hadoop, Hadoop Ecosystem, Reading and Writing Files on Hadoop Distributed File System. Map Reduce basics, Map Reduce Algorithm Design. Graph Algorithms, Data Mining with Big Data, Stream Data Processing, Stream Data Processing System and Tools, Overview of Spark, Spark SQL-Mllib-GraphX, Apache Kafka, Big Data Analytics in Industry Verticals, Operationalizing Basic Data Analytic Methods Using R, Analytics for Unstructured Data.

Learning Resources:

Text Books:

1. Jure Leskovec, Anand Rajaraman, J D Ullman, Mining Massive Datasets, Cambridge University Press, 2nd Edition, 2014.
2. Jimmy Lin and Chris Dyer, Data Intensive Text Processing with Map Reduce, Morgan & Claypool Publishers,1st Edition, 2010.
3.

Reference Books:

1. Bill Franks, Taming The Big Data Tidal Wave, 1st Edition, Wiley, 2012.
2. Johannes Ledolter, Data Mining and Business Analytics with R, Wiley, 2013.

CD17025 **3-0-0 (3)**

Distributed Operating Systems

Pre-Requisites: **None**

Course Outcomes:

CO-1	Analyze the hardware and software issues in modern distributed system
CO-2	Understand distributed architecture, naming, synchronization, consistency and replication, fault tolerance, security and distributed file system
CO-3	Analyze shared memory techniques.
CO-4	Understand various file access methods.
CO-5	Implementation of synchronization and deadlock.

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	3	3	2	3	2	1
CO-2	3	3	2	3	3	1
CO-3	2	2	1	2	2	2
CO-4	2	2	1	2	2	1
CO-5	3	3	2	2	3	1

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Introduction to distributed systems – Communication models in distributed systems – Synchronization – Processor allocation – Scheduling algorithms – Distributed file systems – Distributed shared memory – Case Studies of Distributed systems.

Learning Resources:

Text Books:

1. Andrew S. Tanenbaum, Distributed Operating Systems, 2/e, Pearson Education Asia Publishers, 2002.
2. Ajay D. Kshemakalyani, Mukesh Singhal, DistributedComputing, Cambridge University Press, 2008.

CD17027 **3-0-0 (3)**

Foundations of Data Science

Pre-Requisites: **None**

Course Outcomes:

CO-1	Apply statistical methods to data for inferences.
CO-2	Analyze data using Classification, Graphical and computational methods.
CO-3	Understand Data Wrangling approaches.
CO-4	Perform descriptive analytics over massive data.

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	2	2	3	1	1	1
CO-2	2	3	3	2	2	1
CO-3	2	2	2	2	2	1
CO-4	2	3	3	2	2	1

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Overview of Random variables and distributions, Statistical learning, Assessing model accuracy, Descriptive Statistics, Dependent and Independent events. Linear Regression: Simple and multiple linear regressions, Comparison of Linear regression with K-nearest neighbors. Simple Hypothesis Testing, Student's t-test, paired t and U test, correlation and covariance, tests for association. Classification: Linear and Logistic Regression, LDA and comparison of classification methods Graphical Analysis: Histograms and frequency polygons, Box-plots, Quartiles, Scatter Plots, Heat Maps Programming for basic computational methods such as Eigen values and Eigen vectors, sparse matrices, QR and SVD, Interpolation by divided differences. Data Wrangling: Data Acquisition, Data Formats, Imputation, The split-apply-combine paradigm. Descriptive Analytics: Data Warehousing and OLAP, Data Summarization, Data de-duplication, Data Visualization using CUBEs.

Learning Resources:

Text Books:

1. Gareth James Daniela Witten Trevor Hastie, Robert Tibshirani, An Introduction to Statistical Learning with Applications in R, February 11, 2013,
2. Mark Gardener, Beginning R The Statistical Programming Language, Wiley, 2015.
3. Han, Kamber, and J Pei, Data Mining Concepts and Techniques, 3rd edition, Morgan Kaufman, 2012. (Chapter 2 and Chapter4)

CD17029 **3-0-0 (3)**

Full Stack Development

Pre-Requisites: **CD16009**

Course Outcomes:

CO-1	Understand, analyze and build dynamic and interactive web sites
CO-2	Understand current and evolving Web languages for integrating media and user interaction in both front end and back end elements of a Web site
CO-3	Analysis and reporting of web data using web analytics
CO-4	Applying different testing and debugging techniques and analyzing the web site effectiveness.

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	3	1	3	-	1	2
CO-2	3	1	3	-	1	2
CO-3	3	1	3	-	1	2
CO-4	3	1	3	-	1	2

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

HTML: Introduction to HTML, What is HTML, HTML Documents, Basic structure of an HTML document, creating an HTML document, Mark up Tags, Heading-Paragraphs, Line Breaks, HTML Tags. Elements of HTML: Introduction to elements of HTML, Working with Text, Working with Lists, Tables and Frames, Working with Hyperlinks, Images and Multimedia, Working with Forms and controls. Java Script: Introduction to JavaScript, Basic Syntax, Control Structures, Writing Functions, Working with Arrays, The Document Object Model, Events Handling, Client-side Validation, Form Validation & RegExps

PHP: PHP installation and Introduction, Loops String Functions in PHP, PHP Email Function, PHP Basics, Variables Arrays in PHP with Attributes Date & Time, Image, Uploading File handling in PHP Functions in PHP, Errors handling in PHP.

Python: Introduction to Python, Python basics, Data Types and variables Operators, Looping & Control Structure List, Modules Dictionaries, String Regular Expressions, Functions and Functional Programming, Object Oriented Linux Scripting Environment – Classes, Objects and OOPS concepts, File and Directory Access, Permissions and Controls Socket, Libraries and Functionality Programming, Servers and Clients Web Servers and Client scripting, Exploit Development techniques, Writing plugins in Python, Exploit analysis, Automation Process, Debugging basics, Task Automation with Python

React: Introduction to React, Add React to a website, create a new React App, Introducing JSX, Rendering Elements, Components of Props, State and Lifecycle, Handling events, Conditional Rendering, Lists and keys, Forms, Building single page applications with React

Application using Node JS and MongoDB: Introduction to Node.js; Events; Streams; Modules; Express; Socket.io; Persisting Data. Introduction to MongoDB, Accessing MongoDB from Node JS

Learning Resources:

Text Books:

1. Deitel, Deitel and Nieto, "Internet and Worldwide Web - How to Program", 5th Edition, PHI, 2011.
2. Bai and Ekedhi, "The Web Warrior Guide to Web Programming", 3rd Edition, Thomson, 2008.
3. Brad Dayley, Brendan Dayley, and Caleb Dayley , Node.Js, Mongodb and Angular Web Development: The Definitive Guide to Using the Mean Stack to Build Web Applications, 2nd Edition, –Pearson Education, 2018

Reference Books:

1. Robertw. Sebesta, Programming World Wide Web, 8th Edition –Pearson Education, 2020

CD17031 **3-0-0 (3)**

Information Systems Management

Pre-Requisites: **None**

Course Outcomes:

CO-1	Determine key terminologies and concepts including IT, marketing, management, economics, accounting, finance in the major areas of business.
CO-2	Design, develop and implement Information Technology solutions for business problems.
CO-3	Analysis of computing systems and telecommunication networks for business information systems.
CO-4	Understand ethical issues that occur in business, evaluate alternative courses of actions and evaluate the implications of those actions.
CO-5	Plan projects, work in team settings and deliver project outcomes in time.

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	1	1	2	2	1	1
CO-2	2	3	2	1	2	1
CO-3	2	1	-	1	1	2
CO-4	2	1	1	2	2	1
CO-5	1	2	1	2	2	3

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Organization and Information Systems- Kinds of information systems- System Analysis and Development and Models- Manufacturing and Service Systems - Information systems- Enterprise System- Enterprise Resources Planning- Choice of IT - Nature of IT decision- Security and Ethical Challenges - Ethical responsibilities of Business Professionals.

Learning Resources:

Text Books:

1. Kenneth J Laudon, Jane P.Laudon, Management InformationSystems, 10/e, Pearson/PHI, 2007
2. W. S. Jawadekar, Management Information Systems, 3/e,TMH, 2004

CD17033 **3-0-0 (3)**

Modeling and Simulation

Pre-Requisites: **None**

Course Outcomes:

CO-1	Understand the concept of probability theory
CO-2	Analyze different distributions
CO-3	Understand the concept of Poisson process, Marko chains and Stochastic process
CO-4	Identify different building blocks of simulation and analyze simulation results

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	3	-	-	-	-	-
CO-2	3	1	1	-	-	-
CO-3	3	2	1	-	-	-
CO-4	3	3	2	-	-	2

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Probability theory – Discrete and continuous random variables and distributions - Poisson process - Markov chains – Stochastic process - Building blocks of Simulation - Analysis of Simulation results.

Learning Resources:

Text Books:

1. Sheldon M. Ross, Introduction to Probability Models, 7/e, Academic Press, 2002.
2. Donald E. Knuth, The Art of Computer Programming – Volume 2: Semi Numerical Algorithms, 2/e, Addison Wesley, Reading MA, USA, 2000.
3. Louis G Birta and Gilbert Arbez, Modelling and Simulation: Exploring Dynamic System Behavior, Springer Publishers, 2010.

CD17035 **3-0-0 (3)**

Soft Computing Techniques

Pre-Requisites: **None**

Course Outcomes:

CO-1	Understand the concepts of population based optimization techniques
CO-2	Examine the importance of exploration and exploitation in heuristic optimizationtechniques to attain near-global optimal solution
CO-3	Evaluate the importance of parameters in heuristic optimization techniques
CO-4	Apply for the solution of multi-objective optimization

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	3	2	-	-	-	-
CO-2	3	2	-	1	-	-
CO-3	3	1	-	1	-	-
CO-4	3	1	-	1	-	-

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Fundamentals of Soft Computing Techniques: Definition-Classification of optimization problems-Unconstrained and Constrained optimization Optimality conditions- Introduction to intelligent systems- Soft computing techniques-Classification of meta-heuristic techniques –Single solution based and population based algorithms–Exploitation and exploration in population based algorithms - Properties of Swarm intelligent Systems - Application domain –Discrete and continuous problems- Single objective and multi-objective problems.

Genetic Algorithm and Particle Swarm Optimization: Genetic algorithms- Genetic Algorithm versus Conventional Optimization Techniques-Genetic representations and selection mechanisms; Genetic operators-different types of crossover and mutation operators-Bird flocking and Fish Schooling – anatomy of a particle- equations based on velocity and positions –PSO topologies-control parameters. Application to SINX maximization problem.

Ant Colony Optimization and Artificial Bee Colony Algorithms: Biological ant colony system – Artificial ants and assumptions –Stigmergic communications- Pheromone updating-local-global Pheromone evaporation - ant colony system- ACO models-Touring ant colony system-maxmin ant system - Concept of elistic ants-Task partitioning in honey bees - Balancing foragers and receivers- Artificial bee colony (ABC) algorithms-binary ABC algorithms.

Shuffled Frog-Leaping Algorithm and Bat Optimization Algorithm: Bat Algorithm- Echolocation of bats-Behavior of microbats- Acoustics of Echolocation- Movement of Virtual Bats- Loudnessand Pulse Emission-Shuffled frog algorithm-virtual population of frogs-comparison of memes and genes- memeplex formation-memeplex updation. Application to multi-modal function optimization. Introduction to Multi-Objective optimization- Concept of Pareto optimality.

Learning Resources:

Text Books:

1. Xin-She Yang, “Recent Advances in Swarm Intelligence and Evolutionary Computation, Springer International Publishing, Switzerland, 2015.
2. Kalyan moy Deb, Multi-Objective Optimization using Evolutionary Algorithms, John Wiley & Sons, 2001.
3. James Kennedy and Russel E Eberheart, Swarm Intelligence,The Morgan Kaufmann Series in Evolutionary Computation, 2001.
4. EricBonabeau, MarcoDorigo and Guy Theraulaz, Swarm Intelligence-From natural to Artificial Systems, Oxford university Press, 1999.
5. David Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning, Pearson Education, 2007.
6. Konstantinos E. Parsopoulos and Michael N. Vrahatis, Particle Swarm Optimization and Intelligence: Advances and Applications, Information science reference, IGI Global, 2010.
7. N P Padhy, Artificial Intelligence and Intelligent Systems, Oxford University Press, 2005.

CD17022 **3-0-0 (3)**

Agri-Bioinformatics

Pre-Requisites: **None**

Course Outcomes:

CO-1	Understand the theoretical basis behind Agri informatics and bioinformatics.
CO-2	Compute homologues, analyze sequence alignment and similarity.
CO-3	Understand the Management Information System, Decision support system, Expert system
CO-4	Analyze various gene sequences, gene prediction, and categories of gene prediction.
CO-5	Determine and model biological information and apply to the solution of biological problems.

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	3	-	2	-	3	-
CO-2	-	2	3	-	-	1
CO-3	3	-	2	-	3	-
CO-4	2	3	2	3	2	-
CO-5	-	1	3	-	1	-

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Introduction to Bioinformatics: What is a Data Base, Types of Databases, Biological Databases, Biological Databases, Information Retrieval from Biological Databases

Pair wise Sequence Alignment: Evolutionary Basics, Sequence homology versus similarity, Sequence similarity versus Identity, Scoring Matrices, Statistical Significance of Sequence alignment

Database similarity searching: Unique requirement of Database searching, Heuristic Database searching,

Basic alignment search tool: Comparison of FASTA and BLAST, Multiple Sequence Alignment, Scoring Function, Gene Prediction, Categories of gene prediction, Gene prediction in prokaryotes and Eukaryotes.

Introduction to Agri-informatics: Importance of Agriculture to Development and the Contribution of ICTs, Critical Information Flows and the Agricultural Knowledge System, The Feasibility of ICT in Rural Areas, Financial Aspects: Up-front Costs and Long-term Benefits of ICT Approaches, Technical Aspects of ICT Feasibility in Rural Areas, Promising Emerging Technologies, Some major ICT trends, Some high impact innovations, Hardware and Connectivity, Ubiquitous Telecommunication Infrastructure, Utility or Cloud Computing, Software and Content Management, Interaction with Biology, Biotechnology, Nanotechnology, Materials Science and ICTs, Information and Communication Technology in Agricultural Development, Convergence of ICT with agricultural development, Areas of IT convergence, Drivers of ICT in Agriculture, Low-Cost and Pervasive Connectivity, Adaptable and More Affordable Tools, New Business Models and Public-Private Partnerships, Democratization of Information, the Open Access Movement and Social Media, Use Appropriate Technologies, Making ICT Infrastructure, Appliances and Services, Accessible and Affordable in Rural Areas, Key Challenges and Enablers, Partnerships, Regulation and Policy Challenges, Infrastructure, Local Loop or Last Mile Connectivity, Appliances, Services, Anytime, Anywhere: Mobile

Devices and Services, Applications in Agriculture, Information Systems in Agriculture: Introduction, Information systems and communication networks, Management Information Systems, Internal Information Systems, External Information Systems, Farm Management Information System (FMIS), Farm Management Information System. Decision Support Systems: Introduction, Historical Overview of Decision Support Systems (DSS), Future trends, Decision Support Systems in Agriculture. Expert Systems and Applications: Introduction, what is an Expert System, Expert Systems and Human Reasoning, Structural Nature of Expert Systems, Characteristics of Expert System Applications, Knowledge Acquisition, Knowledge Representation, Explanation, Expert System Strengths and Limitations, Extensions to Expert Systems.

Learning Resources:

Text Books:

1. Jin Xiong, "Essential Bioinformatics", 1st Edition, Cambridge University Press, 2011.
2. Arthur M Lesk, "Introduction to Bioinformatics", 2nd Edition, Oxford University Press, 2007.
3. Zhumur Ghosh “Bioinformatics” 2nd Edition, Oxford University Press, 2009
4. Herdon, Miklos “Agriinformatics” 2nd Edition Cambridge University Press 2013

CD17024 **3-0-0 (3)**

Design Patterns

Pre-Requisites: **None**

Course Outcomes:

CO-1	Understand common design patterns in the context of incremental/iterative development
CO-2	Evaluate and retractor software source code using patterns
CO-3	Analyze and combine design patterns to work together in software design
CO-4	Implement the design patterns in an object oriented language.

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	2	3	2	2	2	1
CO-2	1	2	2	2	3	3
CO-3	2	3	3	2	2	2
CO-4	1	2	2	2	3	2

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Introduction: What Is a Design Pattern, Design Patterns in Smalltalk MVC, Describing Design Patterns, the Catalog of Design Patterns, Organizing the Catalog, How Design Patterns Solve Design Problems, How to Select a Design Pattern, How to Use a Design Pattern.

A Case Study: Designing a Document Editor: Design Problems, Document Structure, Formatting, Embellishing the User Interface, and Supporting Multiple Look-and-Feel Standards, Supporting Multiple Window Systems, User Operations, Spelling Checking and Hyphenation.

Creational Patterns: Abstract Factory, Builder, Factory Method, Prototype, Singleton.

Structural Pattern: Adapter, Bridge, Composite, Decorator, Façade, Flyweight, Proxy.

Behavioral Patterns: Chain of Responsibility, Command, Interpreter, Iterator, Mediator, Memento, Observer, State, Strategy, Template Method, Visitor, a Brief History, and the Pattern Community

Learning Resources:

Text Books:

1. Erich Gamma, Design Patterns, Addison-Wesley, 1994.
2. Frank Buschmann, Regine Meunier, Hans Rohnert, Peter Sommerlad, Michael Stal, Pattern-Oriented Software Architecture: A System of Pattern, John Wiley & Sons; 1996

CD17026 **3-0-0 (3)**

ERP and Supply Chain Management

Pre-Requisites: **None**

Course Outcomes:

CO-1	Understand the important role of ERP and Supply chain management
CO-2	Implement ERP and analyze its product life cycle
CO-3	Apply supply chain theories, practices and concepts utilizing case problems and problem based learning situations
CO-4	Analyze supply chain modelling strategies for policy management

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	-	2	1	1	2	3
CO-2	-	2	1	-	-	1
CO-3	-	2	2	-	-	1
CO-4	-	2	1	1	2	3

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Strategizing ERP - Customer Relationship Management - ERP implementation - Product life cycle management- Introduction to Supply Chain Management - Supply chain process management - policy management - quality and deployment - modeling the strategic supply chain - Decision science models for supply chain excellence.

Learning Resources:

Text Books:

1. Christian N Madu, ERP and Supply Chain Management, Chi Publishers, 2005.
2. Marianne Bradford, Modern ERP: Select, Implement and Use Today's Advanced Business Systems, Lulu Press Inc., 2/e, 2010.

CD17028 **3-0-0 (3)**

Human Computer Interaction

Pre-Requisites: **None**

Course Outcomes:

CO-1	Understand the usability of interactive systems.
CO-2	Apply the techniques to manage the design process.
CO-3	Use the appropriate interaction style for a given problem.
CO-4	Design an interface for a given scenario, based on the concepts of HCI.

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	1	1		1	1	1
CO-2	2	2		2	2	2
CO-3	2	2		2	2	2
CO-4	3	3		3	3	2

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Introduction: Usability of Interactive Systems- Usability goals and measures, usability motivations, universal usability, goals for Professional

Managing design processes: Introduction, Organizational design to support usability, Four pillars of design, Development methodologies, Ethnographic observation, Participatory design, Scenario Development, Social impact statement for early design review, legal issues, Usability Testing and Laboratories.

Design: Introduction, Task- Related Menu Organization, Single menus, Combinations of Multiple Menus, Content Organization, Fast Movement Through Menus,

Data entry with Menus: Form Fill-in, dialog Boxes, and alternatives, Audio Menus and menus for Small Displays.

Command and Natural Languages: Command organization Functionality, Strategies and Structure, Naming and Abbreviations, Natural Language in Computing

Interaction Devices: Introduction, Keyboards and Keypads, Pointing Devices, Speech and Auditory Interfaces, Displays- Small and large.

Learning Resources:

Text Books:

1. Ben Shneiderman, Catherine Plaisant, Maxine Cohen, Steven M Jacobs, Designing the User Interface, Strategies for Effective Human Computer Interaction, 5ed, Pearson
2. Wilbert O Galitz, The Essential guide to user interface design, 2/e, Wiley DreamaTech.

CD17030 **3-0-0 (3)**

Image Processing

Pre-Requisites: **None**

Course Outcomes:

CO-1	Classify Image representations
CO-2	Apply Image transformation methods
CO-3	Implement image processing algorithms
CO-4	Design object detection and recognition algorithms
CO-5	Recover the information, knowledge about the objects in the scene

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	2	3	3	2	1	1
CO-2	2	1	2	2	1	1
CO-3	1	2	2	2	1	1
CO-4	2	3	2	2	1	1
CO-5	1	2	2	2	1	1

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Introduction to Image Processing, Image Processing Applications, Image Representation and Modeling, Image Acquisition, Image Enhancement, Image Filtering, Edge Detection, Segmentation, ImageCompression, different Image Transformations, Object recognition, Color Image Processing, Wavelets, Texture, Feature Extraction.

Learning Resources:

Text Books:

1. Gonzalez and Woods, Digital Image Processing, 3/e, PrenticeHall, 2007.
2. M Sonka et. Al., Image Processing: Analysis and MachineVision, 3/e, Cole Pub. Co., 2008

CD17032 **3-0-0 (3)**

Mobile Computing

Pre-Requisites: **None**

Course Outcomes:

CO-1	Identify mobile computing societal applications and communication constraints in wireless environment
CO-2	Analyze mobile- IPv4 and IPv6 architectures with agents and proxies.
CO-3	Design MAC protocols for wireless networks.
CO-4	Evaluate the performance of TCP protocols in Wireless Networks with mobile nodes.

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	1	1	1	1	1	1
CO-2	2	2	-	2	2	1
CO-3	1	2	-	2	3	1
CO-4	1	1	-	2	2	1

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Basic communication Technologies, Introduction to Mobile Networks, Introduction to different categories of Wireless networks (MANET: Mobile ad-hoc networks-Communication Architectures of a typical MANET, Applications of MANET, WSN: Wireless Sensor Networks-topologies in WSN- Linear, Grid and Cluster based topologies, communication architectures in a WSN, applications of WSNs, VANET: Vehicular Ad-hoc Networks- communication architectures in VANET, Applications of VANET, PAN: Personal Area Networks- the Bluetooth technology, the blue tooth specifications, DTN: Delay Tolerant Network-delay tolerant network architecture, applications of DTN), Wireless Communication Fundamentals, Cellular Wireless Networks.

Medium Access Control Layer- Hidden terminal problem, Exposed terminal problem, Collision avoidance, Congestion Avoidance, Congestion control, Energy Efficiency, MACA and MACAW protocols, Wireless LAN and IEEE 802.11- Network architecture, the physical layer, the MAC layer, security.

Detailed network layer functionalities in multi-hop wireless networks- Mobile Ad-hoc Networks- broadcasting in a MANET, flooding generated broadcasts torm problem, rebroadcasting schemes, Issues in providing multicasting in MANET, Multicast routing protocols, Geocasting-Geocastrouting protocols. Mobile Network Layer(MobileIP), DHCP(Dynamic host configuration protocol), Routing in Mobile Ad hoc Networks (MANET)- Topology-based versus position based approaches, Proactive routing protocols, Reactive routing protocols, Hybrid routing protocols, position based routing issues and forwarding strategies, AODV (Ad-hoc On-Demand Distance Vector Routing Protocol)- Analysis of AODV under mobility and Faults in a network, DSR(Dynamic Source Routing)-Analysis of DSR under mobility and Faults in a network, Secure routing protocols in MANET, Wireless Sensor Networks: (Routing protocols, Localization methods, Sensor Deployment Strategies), traffic flow pattern in WSN- one to many, many to one and many to many, Routing protocols for Delay Tolerant Networks, Routing protocols for Vehicular Ad-hoc Networks, Wireless Access Protocol, GPS (Global positioning system) and applications, RFID and its applications.

Learning Resources:

Text Books:

1. Jochen Schiller, Mobile Communications,Second Edition, Pearson Education, 2003.
2. CDMCordeiro, D.P.Agarwal, Adhoc and Sensor Networks:Theory and applications, World Scientific, 2006.

CD17034 **3-0-0 (3)**

Natural Language Processing

Pre-Requisites: **None**

Course Outcomes:

CO-1	Understand the Text representation and Text pre-processing techniques
CO-2	Understand language modeling with N-Grams.
CO-3	Apply syntactic parsing to produce parse trees
CO-4	Design NLP Systems for Text Summarization, Classification and Translation
CO-5	Evaluate the performance NLP System

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	1	1	-	-	1	–
CO-2	1	2	1	-	2	–
CO-3	1	2	2	3	1	–
CO-4	2	3	3	1	3	–
CO-5	-	-	-	-	2	–

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Introduction and Basic Text Processing, Text Representation- one-hot encoding, TF-IDF, Bag-of-Words, Word2Vec, Glove Embedding, Text-Preprocessing- Tokenization, Stemming-Porters Stemming algorithm, Lemmatization, Normalization, Spell Checker- Spelling Correction, Edit Distance, Language Modeling- Probabilistic Language Modelling- N-gram Modelling, Language Models Evaluation, Text Parsing, Part-of-Speech Tagging, POS with Hidden Markov model (HMM), Text Summarization-Abstractive and Extractive Text Summarization. Text Classification- Text Classification Techniques – Topic Modelling, Sentiment Analysis, Named Entity Recognition(NER), Machine Translation, Introduction to Large Language Models (LLMs)

Learning Resources:

Text Books:

1. Daniel Jurafsky and James H. Martin, "Speech and Language Processing", 3rd Edition, 2008.
2. Uday Kamath, John Liu, James Whitaker, “Deep Learning for NLP and Speech Recognition”, Springer, 2020

Reference Books:

1. Allen, James, "Natural Language Understanding", Second Edition, Benjamin/ Cumming, 1995.
2. Sowmya Vajjala, Bodhisattwa Majumder, Anuj Gupta, Harshit Surana, “Practical Natural Language Processing: A Comprehensive Guide to Building Real-World NLP Systems”, O’REILY, 2020

CD17036 **3-0-0 (3)**

Software Architecture

Pre-Requisites: **CD16001 and CD16005**

Course Outcomes:

CO-1	Apply the theory of Software Architecture.
CO-2	Examine existing Architectural styles
CO-3	Learning different techniques
CO-4	Practicing different software architecture tools

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	2	2	2	2	2	2
CO-2	2	2	2	2	3	2
CO-3	3	3	3	3	3	2
CO-4	2	2	2	2	3	2

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Introduction and architectural drivers, Introduction –Architecture Standard Definitions – Architectural structures – Influence of software architecture on organization-both business and technical – Architecture Business Cycle- Introduction – Functional requirements – Technical constraints – Quality

Quality Attribute Workshop – Documenting Quality Attributes – Six part scenarios – Case studies.

Architectural views - Introduction – Standard Definitions for views – Structures and views – Representing views-available notations – Standard views – 4+1 view of RUP, Siemens 4 views, SEI's perspectives and views – Case studies

Architectural styles - Introduction – Data flow styles – Call-return styles – Shared Information styles – Event styles – Case studies for each style.

Documenting the architecture Good practices – Documenting the Views using UML – Merits and Demerits of using visual languages – Need for formal languages – Architectural Description Languages – ACME – Case studies. Special topics: SOA and Web services – Cloud Computing – Adaptive structure.

Learning Resources:

Text Books:

1. Len Bass, Paul Clements, and Rick Kazman, Software Architectures Principles and Practices, 2nd Edition, Addison-Wesley, 2003.
2. Anthony J Lattanze, Architecting Software Intensive System. A Practitioner's Guide, Auerbach Publications, 2010.
3. Paul Clements, Felix Bachmann, Len Bass, David Garlan, James Ivers, Reed Little, Paulo Merson, Robert Nord, and Judith Stafford, Documenting Software Architectures. Views and Beyond, 2nd Edition, Addison-Wesley, 2010.

4. Paul Clements, Rick Kazman, and Mark Klein, Evaluating software architectures: Methods and case studies, Addison-Wesley, 2001.
5. Rajkumar Buyya, James Broberg, and Andrzej Goscinski, Cloud Computing. Principles and Paradigms, John Wiley & Sons, 2011

CD18021 **3-0-0 (3)**

Cloud Computing

Pre-Requisites: **None**

Course Outcomes:

CO-1	Determine Cloud Computing Architectures and Services for various societal applications
CO-2	Analyze Cloud infrastructure including Google Cloud and Amazon Cloud.
CO-3	Develop private and hybrid cloud for organizations to execute customized applications
CO-4	Analyze authentication, confidentiality and privacy issues in Cloud computing environment.
CO-5	Determine financial and technological implications for selecting cloud computing platforms

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	3	3	2	3	3	1
CO-2	2	2	-	3	2	-
CO-3	3	3	-	3	3	1
CO-4	1	1	1	1	3	1
CO-5	3	3	-	2	2	1

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Distributed System Models and Enabling Technologies, Computer Clusters for Scalable Parallel Computing, Virtual Machines and Virtualization of Clusters and Data Centres, Cloud Platform Architecture over Virtualized Data Centres, Service-Oriented Architectures for Distributed Computing, Cloud Programming and Software Environments, Grids, P2P and the Future Internet, Ubiquitous Clouds and the Internet of Things, Advanced Topics in Cloud Computing

Learning Resources:

Text Books:

1. Kai Hwang, Geoffrey C. Fox and Jack J. Dongarra, Distributed and Cloud Computing: From Parallel Processing to the Internet of Things, Morgan Kaufmann, Elsevier, 2012.
2. Anothony T Velte, Toby J Velte, Robert Elsenpeter, Cloud Computing: A Practical Approach, MGH, 2010.
3. Gautam Shroff, Enterprise Cloud Computing, Cambridge, 2010
4. Ronald Krutz and Russell Dean Vines, Cloud Security, 1/e, Wiley, 2010

CD18023 **3-0-0 (3)**

Deep Learning

Pre-Requisites: **None**

Course Outcomes:

CO-1	Identify Convolutional Neural Networks models to solve Supervised Learning Problems
CO-2	Design Auto encoders to solve Unsupervised Learning problems
CO-3	Apply Long Shot Term Memory (LSTM) Networks for time series analysis classification problems.
CO-4	Apply Classical Supervised Tasks for Image Denoising, Segmentation and Objectdetection problems.

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	2	2	3	2	2	1
CO-2	2	2	3	2	2	1
CO-3	2	2	3	2	2	1
CO-4	2	2	3	2	2	1

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

INTRODUCTION–History of Deep Learning, Introduction to Neural Network, Perceptrons, Perceptron Learning Algorithm. Multilayer Perceptrons (MLPs), Representation of MLPs, Sigmoid Neurons, Gradient Descent. Feed Forward Neural Networks, Backpropagation. Gradient Descent (GD), Momentum Based GD, Stochastic GD; REGULARIZATION

Bias Variance Tradeoff, L2regularization, Earlystopping, Dataset augmentation, Parameter sharing and tying; SUPERVISED DEEP LEARNING-Convolutional Neural Networks, Building blocks of CNN, Transfer Learning, LeNet, AlexNet, ZFNet, VGGNet, GoogLeNet, ResNetModels, Visualizing Convolutional Neural Networks; Unsupervised Learning with Deep Network, Autoencoders,

Variational Autoencoder, Regularization in autoencoders, Denoising autoencoders, Sparse autoencoders; Recent Trends in Deep Learning Architectures, Residual Network, Skip Connection Network, Fully Connected CNN, Recurrent Neural Networks(RNN), Long Shot Term Memory (LSTM) Networks, Generative Adversarial Networks(GAN); Classical Supervised Tasks with Deep Learning, Image Denoising, Semantic Segmentation, Object Detection.

Learning Resources:

Text Books:

1. Ian Good Fellow, Yoshua Benjio, Aaron, Deep Learning Courville, The MIT Press.
2. Christopher Bishop, Pattern Recognition and Machine Learning, Springer, 2006.

CD18025 **3-0-0 (3)**

Digital Forensics

Pre-Requisites: **None**

Course Outcomes:

CO-1	Understand the need for digital forensics
CO-2	Identify different technologies for digital forensics
CO-3	Understand different investigation methodologies
CO-4	Apply the digital forensics for different fields.

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	2	2	1	2	3	2
CO-2	1	3	2	2	2	2
CO-3	1	2	1	2	2	2
CO-4	1	2	1	2	2	2

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Information formats, PC hardware, Disc geometry, File system, Electronic organizers. Forensic analysis, Investigative Methodology: Forensic Analysis, Electronic Discovery, Intrusion Investigation. Technology: Windows Forensic Analysis, UNIX Forensic Analysis, Embedded Systems Analysis, Mobile Network Investigations. Intrusion Investigation, Analysis tools, financial forensics.

Learning Resources:

Text Books:

1. Sammes T, B.Jenkinson, Forensic Computing, Springer, 2007.
2. Eoghan Casey.Ed., Handbook of Digital Forensics and Investigation, Academic Press, 2010.

CD18027 **3-0-0 (3)**

E-Commerce Technologies and Management

Pre-Requisites: **None**

Course Outcomes:

CO-1	Understand E-commerce frameworks and advantages
CO-2	Acquire in-depth knowledge of e-commerce business models and managing an enterprise
CO-3	To demonstrate clear, concise, thoughtful and good understanding of electronic payment system and its types
CO-4	To demonstrate a good understanding of e-marketing and its types
CO-5	Comprehend the future trends in e-commerce

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	1	3	2	3	3	2
CO-2	2	3	2	3	3	2
CO-3	2	3	3	2	3	2
CO-4	2	3	3	2	3	1
CO-5	1	2	2	2	2	2

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Overview – E-commerce Infrastructure – Wireless Technology – Web Architecture – Data interchange – Web content delivery – Access Security – Public Key Encryption – Electronic Payment System – Mass Personalization – Search Engines – Data Mining and Privacy – Intelligent Agents – Auction Models

Learning Resources:

Text Books:

1. Kenneth C Louden, E-Commerce: Business, Technology,Society, 7/e, PHI, 2011.
2. Turban, Electronic Commerce 2010: A ManagerialPerspective, 6/e, Pearson, 2010

CD18029 **3-0-0 (3)**

Foundations of Blockchain

Pre-Requisites: **CD16008, CD16010 and CD17004**

Course Outcomes:

CO-1	To understand the essential concepts and structural components of blockchain and the rationale behind its implementation.
CO-2	To understand the core technology, many types of blockchains and protocols that operate the blockchain.
CO-3	To articulate and develop and test blockchain-compatible diverse applications with smart contracts.
CO-4	To understand and analyze the advantages and disadvantages of employing blockchain technology in various industries and technologies.

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	2	2	2	2	-	-
CO-2	2	2	2	2	2	2
CO-3	3	3	3	3	3	3
CO-4	3	3	2	2	2	-

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Introduction to Blockchain- Key Concepts of Blockchain, Features of Blockchain, Importance of Blockchain, Blockchain 1.0, 2.0, and 3.0, Issues to Centralized System, Centralized to Decentralized and Distributed System, Building Blocks of Blockchain- Distributed Ledgers & P2P Networks, Block Header, Transaction Organization.

Cryptographic Primitives, Basic Crypto Primitives- Hash Functions- Properties of Hash Function, Nonce, Merkle Trees, Hash Pointers, Public Key Cryptography- Public/private keys, Signature schemes, Signature correctness,

Decentralization- Distributed shared ledger, Distributed Consensus-Distributed Consensus Protocol. Classical theory of consensus, Byzantine Generals Problem possibility and impossibility results, Asynchronous consensus and Byzantine Fault Tolerance

Types of Blockchain- Permission-less Blockchain- Bitcoin-Introduction to Bitcoin, Bitcoin Transaction, Bitcoin Protocol, Bitcoin Wallets, Bitcoin Block, Bitcoin Scripts, Bitcoin Network, Bitcoin Mining-Nakamoto Consensus- Proof-of-work, Mining target T, Proof-of-work equation, Mining Algorithm, Mining and reward, Block freshness, Partial and full nodes, Attacks on Bitcoin- Double-spend attacks, Selfish mining, Ethereum Blockchain, Introduction to Ethereum, Ethereum Networks, Ethereum Wallets, Ethereum Clients, Ethereum accounts, Transactions and State, Smart contracts, Privacy-preserving smart contracts, Proof-of-stake, Variants of Ethereum blockchain

Permissioned Blockchain- Hyperledger Fabric-, State Machine Replication, Distributed State Machines, MSP, Consensus- Raft Consensus Algorithm, Safety and liveness, Privacy based blockchain, ZCash, Zero-knowledge-proof, R3 Corda, Corda Network

Blockchain Security- Attacks on Blockchain and their Countermeasures, Application & Use cases of Blockchain- IoT Security using BC, Edge Security using BC, Cloud Security using BC and E2E IoT Security using BC

Learning Resources:

Text Books:

1. Bitcoin and Cryptocurrency Technologies, A. Narayanan, J. Bonneau, E. Felten, A. Miller and S. Goldfeder, Princeton University Press. Henceforth termed as PUP (Princeton university press).
2. Mastering Blockchain: A deep dive into distributed ledgers, consensus protocols, smart contracts, DApps, cryptocurrencies, Ethereum, and more, 3rd Edition, Imran Bashir, Packt Publishing, 2020, ISBN: 9781839213199.
3. Introduction to Cryptocurrencies, a basic online course by Haseeb Qureshi

Reference Books:

1. William Magnuson, “Blockchain Democracy- Technology, Law and the Rule of the Crowd”, Cambridge University Press, 2020.
2. Pethuru Raj, Kavita Saini, Chellammal Surianarayanan, “Blockchain Technology and Applications”, CRC Press, 2021.
3. Chandramouli Subramanian, “Blockchain Technology”, Universities Press, 2020.
4. Relevant Research Paper and While Papers.

CD18031 **3-0-0 (3)**

Game Theory

Pre-Requisites: **None**

Course Outcomes:

CO-1	Analyze games based on complete and incomplete information about the players
CO-2	Analyze games where players cooperate
CO-3	Compute Nash equilibrium
CO-4	Apply game theory to model network traffic

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	1	1	-	1	1	1
CO-2	1	2	-	2	2	2
CO-3	1	2	-	1	2	2
CO-4	1	1	-	2	2	3

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Noncooperative Game Theory: Games in Normal Form - Preferences and utility, examples of normal-form, Analyzing games: Pareto optimality, Nash equilibrium, Maxmin and minmax strategies, dominated strategies, Rationalizability, Correlated equilibrium

Computing Solution Concepts of Normal-Form Games: Computing Nash equilibria of two- player, zero-sum games, Computing Nash equilibria of two-player, general-sum games, Complexity of computing Nash equilibrium, Lemke–Howson algorithm, Searching the space of supports, Computing Nash equilibria of n-player, general-sum games, Computing maxmin and minmax strategies for two- player, general-sum games, Computing correlated equilibria

Games with the Extensive Form: Perfect-information extensive-form games, Subgame-perfect equilibrium, Computing equilibria, Imperfect-information extensive-form games, Sequential equilibrium

Other Representations: Repeated games: Finitely repeated games, Infinitely repeated games, automata, Stochastic games Bayesian games: Computing equilibria

Coalitional Game Theory: Transferable Utility, Analyzing Coalitional Games, Shapley Value, the Core

Mechanism Design: Strategic voting, unrestricted preferences, Implementation, quasilinear setting, efficient mechanisms, and Computational applications of mechanism design, Task scheduling, Bandwidth allocation in computer networks

Auctions: Single-good auctions, Canonical auction families, Bayesian mechanisms, Multiunit auctions, combinatorial auctions,

Learning Resources:

Text Books:

1. Shoham,Y. and Leyton Brown, K. Multiagent Systems:Algorithmic,Game Theoretic, and Logical Foundations.Cambridge University Press, 2008.
2. Osborne, M.J., and Rubinstein, A. A Course in Game Theory. Cambridge, MA: MITPress, 1994.
3. D.Fudenberg and J.Tirole, Game Theory, The MIT Press, 2005

CD18033 **3-0-0 (3)**

GPU Programming

Pre-Requisites: **None**

Course Outcomes:

CO-1	Analyze for the performance of GPU memory hierarchy
CO-2	Develop parallel programs using OpenCL library
CO-3	Generate parallel programs for matrix, graph and sorting problems using Cuda library
CO-4	Compare the performance of different algorithms for the numerical and data processing problems on GPGPUs and suggest methods for improving the performance.

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	2	2	1	2	2	1
CO-2	2	3	1	2	2	1
CO-3	2	3	1	2	2	1
CO-4	2	2	1	1	2	1

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

GPU Computing - Introduction : Introduction to General Purpose Computing on Graphic Processing Units (GPGPU); GPU as parallel computers – CUDA enabled NVIDIA GPUs; AMD- ATI-OpenCL, GPGPU Architecture of a Modern GPU – Threaded Streaming Multi-processors; communication bandwidth; Unified Graphics and Computing Processors; GPGPU- GPU computing – Scalable GPUs; Speed-up & Parallelism; CPU/GPU programing; SPMD programming model

CUDA APIs & CUDA Threads - GPUs-Data Parallelism; GPU-CUDA Program Structure; GPU device memories & Data transfer; Kernel functions and threading; CUDA Runtime API; CUDA Thread Execution; CUDA Thread organization; Synchronization; Thread Scheduling;

CUDA Memory and Performance Considerations : GPUs-Memory Access Efficiency; CUDA Device Memory types; CUDA memory model – constant memory; shared memory; local memory; global memory – Performance Issues; Unified Address space- NVIDIA GPUS; Global Memory Bandwidth; Thread Granularity; Memory Coalescing; Using Multiple GPUs; CUDA – matrix into matrix multiplication using shared memory without shared memory

Performance Issues - Matrix Computations : Performance Considerations; Data Prefetching; Shared memory resources; Programming on Dense Matrix computations (Vector-Vector Multiplication; Matrix-Vector Multiplication; Matrix-Matrix Multiplication.

OpenCL (Open Computing Language) : Heterogeneous Computing – Programming; Data Parallelism Model – OpenCL; OpenCL, Device Architecture; OpenCL Kernel Functions; OpenCL APIs – Matrix- Matrix, Computations using different partitioning techniques– OpenCL; OpenCL – Device Management and Kernel launch; Compilation Model and programming features of OpenCL – Device query; Object Query, and task parallelism model

Learning Resources:

Text Books:

1. Benedict R Gaster, Lee Howes, David R Kaeli Perhaad Mistry Dana Schaa, Heterogeneous Computing with OpenCL, MGH, 2011
2. Jason Sanders, Edward Kandrot, CUDA By Example – An Introduction to General-Purpose GPU Programming, Addison Wesley, 2011

CD18035 **3-0-0 (3)**

Information Security

Pre-Requisites: **None**

Course Outcomes:

CO-1	Evaluate the risks and vulnerabilities in protocols/Standards.
CO-2	Apply Number Theory and Algebra required for designing cryptographicalgorithms
CO-3	Design symmetric key and asymmetric key encryption techniques.
CO-4	Design authentication, message integrity and authenticated encryption protocols.
CO-5	Design and analyze security of systems

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	2	1	-	1	1	1
CO-2	3	1	-	1	1	1
CO-3	2	2	-	2	2	1
CO-4	2	3	-	2	2	1
CO-5	2	3	-	2	2	1

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Information Security-Security Problems in Databases – Security Controls – Database Security – Security Models – Secure DBMS Design – Design of Secure Databases – Statistical Database Security- Intrusion Detection – Expert systems-based approach – MIDAS – Security Models for next generation Databases – SORION Model

Learning Resources:

Text Books:

1. Silvana Castano, Database Security, Addison Wesley andACM, 1995.
2. Merkov, Information Security: Principles and Practices, 1/e,Pearson, 2007.

CD18037 **3-0-0 (3)**

Malware Detection and Mitigation

Pre-Requisites: **None**

Course Outcomes:

CO-1	Possess the skills necessary to carry out independent analysis of modern malware samples using both static and dynamic analysis techniques.
CO-2	Have an intimate understanding of executable formats, Windows internals and API, and analysis techniques.
CO-3	Extract investigative leads from host and network based indicators associated with a malicious program
CO-4	Apply techniques and concepts to unpack, extract, decrypt, or by pass new anti-analysis techniques in future malware samples.

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	1	2	2	2	3	2
CO-2	2	3	1	2	2	2
CO-3	1	2	1	2	2	2
CO-4	1	2	1	2	2	2

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Introduction to malware, OS security concepts, malware threats, evolution of malware, malware types- viruses, worms, rootkits, Trojans, bots, spyware, adware, logic bombs, malware analysis, static malware analysis, dynamic malware analysis.

Advanced Static Analysis

X86 Architecture- Main Memory, Instructions, Opcodes and Endianness, Operands, Registers, Simple Instructions, The Stack, Conditionals, Branching, Rep Instructions, C Main Method and Offsets. Analyzing Windows programs, Portable executable file format, disassembling malicious executable programs. Anti-static analysis techniques- obfuscation, packing, metamorphism, polymorphism.

Advanced Dynamic Analysis

Debugging malware- ollydbg, windbg, setting virtual environments- sandboxes, emulators, Hypervisors, virtual machines, live malware analysis, dead malware analysis, analyzing traces of malware- system-calls, api-calls, registries, network activities. Anti-dynamic analysis techniques-anti- vm, runtime-evasion techniques.

Malware Functionality

Downloaders, Backdoors, Credential Stealers, Persistence Mechanisms, Privilege Escalation, Covert malware launching- Launchers, Process Injection, Process Replacement, Hook Injection, Detours, APC injection.

Malware Detection Techniques

Signature-based techniques: malware signatures, packed malware signature, metamorphic and polymorphic malware signature. Non-signature based techniques: similarity-based techniques, machine-learning methods, invariant-inferences.

Learning Resources:

Text Books:

1. Sikorski, Michael, and Andrew Honig. Practical malware analysis: the hands-on guide to dissecting malicious software. no starch press, 2012.
2. Filiol, Eric. Computer viruses: from theory to applications. Springer Science & Business Media, 2006.
3. Ligh, Michael, Steven Adair, Blake Hartstein, and Matthew Richard. Malware analyst's cookbook and DVD: tools and techniques for fighting malicious code. Wiley Publishing, 2010.

CD18039 **3-0-0 (3)**

Network Programming

Pre-Requisites: **None**

Course Outcomes:

CO-1	Understand the advanced knowledge of networking
CO-2	Understand advanced knowledge of programming for network communications
CO-3	Analyze various solutions to perform inter-process communication
CO-4	Understand the implication of security issues in networking

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	3	2	1	3	2	1
CO-2	2	2	1	3	2	1
CO-3	2	1	1	2	2	1
CO-4	2	2	2	3	3	1

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Introduction to Network programming - communication protocols - OSI protocols - Protocol comparisons- Introduction to TCP/IP- Elementary TCP and UDP sockets-Advanced sockets- IPV4, IPv6 and Network Interfaces-Programming with HTTP for the Internet- E-mail Protocols, FTP, and CGI Programming - Screen-scraping and Other Practical Applications - Web Services - Network Security

Learning Resources:

Text Books:

1. W. Richard Stevens, Bill Fenner, Andrew M. Rudoff, UNIX® Network Programming Volume 1, Addison Wesley, 2004
2. Brandon Rhodes, John Goerzen, Foundations of Python Network Programming, Apress, 2010
3. M. O. Faruque Sarker, Python network programming cook book, Packt Publishing, 2014

CD18041 **3-0-0 (3)**

Program Analysis and Verification

Pre-Requisites: **None**

Course Outcomes:

CO-1	Apply the theory of abstract interpretation.
CO-2	Examine existing techniques
CO-3	Combine algorithms for program analysis
CO-4	Experiment with Soot and Java software packages.

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	2	2	2	2	2	2
CO-2	2	2	2	2	3	2
CO-3	3	3	3	3	3	2
CO-4	2	2	2	2	3	2

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Introduction to Program Analysis: Overview of Program Analysis, Importance and Applications, Basic Concepts and Terminologies.

Static Analysis: Flow Analysis, Control Flow Analysis (CFA), Data Flow Analysis (DFA), Type Systems, Abstract Interpretation, Symbolic Execution.

Dynamic Analysis: Instrumentation Techniques, Profiling, Monitoring, Testing and Debugging.

Model Checking: Temporal Logic (LTL, CTL), State-Space Exploration, Model Checking Algorithms, Tools for Model Checking (e.g., SPIN, NuSMV).

Program Verification: Hoare Logic, Invariants, Pre-conditions and Post-conditions, Loop Invariants, Theorem Proving.

Formal Methods: Introduction to Formal Methods, Specification Languages (e.g., Z, VDM), Automated Theorem Provers (e.g., Z3, Coq).

Advanced Topics in Static Analysis: Interprocedural Analysis, Context-Sensitive Analysis, Path-Sensitive Analysis, Shape Analysis.

Abstract Interpretation: Principles of Abstract Interpretation, Abstract Domains, Fixed Point Computation, Applications and Case Studies.

Automated Deduction: SAT Solvers, SMT Solvers, Decision Procedures.

Software Model Checking: Bounded Model Checking, Counterexample-Guided Abstraction Refinement (CEGAR), Software Verification Tools (e.g., SLAM, CBMC).

Static Analysis Tools: Introduction to Common Tools, Practical Usage and Case Studies, Integration into Development Workflows.

Security Analysis: Program Security Basics, Static and Dynamic Security Analysis, Vulnerability Detection.

Concurrency Analysis: Analysis of Concurrent Programs, Data Race Detection, Deadlock Detection.

Program Slicing: Techniques for Program Slicing, Applications in Debugging and Maintenance.

Case Studies and Practical Applications: Real-World Examples, Industry Case Studies, Application of Theoretical Concepts.

Trends and Research Directions: Current Trends in Program Anal

Learning Resources:

Text Books:

1. "Principles of Program Analysis" by Flemming Nielson, Hanne R. Nielson, and Chris Hankin
2. "Software Model Checking" by Edmund M. Clarke, Orna Grumberg, and Doron A. Peled
3. "Abstract Interpretation: A Unified Lattice Model for Static Analysis of Programs by Construction or Approximation of Fixpoints" by Patrick Cousot and Radhia Cousot
4. "The Calculus of Computation: Decision Procedures with Applications to Verification" by Aaron R. Bradley and Zohar Manna
5. "Logic in Computer Science: Modelling and Reasoning about Systems" by Michael Huth and Mark Ryan
6. "The Temporal Logic of Reactive and Concurrent Systems: Specification" by Zohar Manna and Amir Pnueli
7. "Static Program Analysis" by Anders Møller and Michael I. Schwartzbach

CD18043 **3-0-0 (3)**

Quantum Computing

Pre-Requisites: **CD16001 and CD16008**

Course Outcomes:

CO-1	Understand quantum computing principles.
CO-2	Design and analyze quantum circuits and algorithms.
CO-3	Implement quantum circuits and algorithms using Qiskit.
CO-4	Understand fundamentals of quantum communication.

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	2	2	–	1	2	–
CO-2	2	3	–	2	3	–
CO-3	2	3	–	2	2	–
CO-4	2	2	–	2	2	–

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Preliminaries: Review of linear algebra and complex numbers.

Quantum Computation: Introduction to qubits, Multiple qubits, Dirac notation, Bloch sphere, Reversable Gates, Basic single- qubit gates, Two-qubit gates, Measurements, Quantum circuits, Bell state circuit, No-cloning theorem, Teleportation, Amplitude Amplification, Superdense coding, Physical realizations of qubits and Qiskit.

Quantum Algorithms: Introduction to query complexity, Deutsch algorithm, Deutsch-Josza algorithm, Bernstein-Vajirani algorithm, Simon`s algorithm, Quantum Fourier Transform, Quantum Phase Estimation, Grover`s Search Algorithm, Ordering finding using phase estimation, Shor`s algorithm, Quantum key distribution, BB84 protocol and HHL algorithm.

Learning Resources:

Text Books:

1. Michael A. Nielsen and Issac L. Chuang, Quantum Computation and Quantum Information,. Cambridge, 2010, 10th Anniversary Edition.
2. Noson S. Yanofsky and Mirco A. Mannucci, Quantum Computing for Computer Scientists, Cambridge University Press, 2008, 1st Edition.

Reference Books:

1. Eleanor G. Rieffel and Wolfgang H. Polak, Quantum Computing: A Gentle Introduction, MIT Press, 2011.
2. Quantum Computing in Practice with Qiskit(R) and IBM Quantum Experience(R): Practical recipes for quantum computer coding at the gate and algorithm level with Python, Hassi Norlén, Packt Publishing, 2020.

Other Suggested Readings:

1. IBM Quantum Learning Courses: https://learning.quantum.ibm.com/ .
2. YouTube Link: Introduction to Quantum Computing Complete Course - Quantum Soar.

CD18045 **3-0-0 (3)**

Reinforcement Learning

Pre-Requisites: **CD16008 and CD17006**

Course Outcomes:

CO-1	Understand elements of RL and solutions to Multi-Arm Bandit Problem.	
CO-2	Formulate and solve problems modelled with Markov Reward Process.	
CO-3	Comprehend solutions for problems with Markov Decision Process.	
CO-4	Apply Dynamic Programming for Markov Decision Process	
CO-5	Analyze Temporal Difference Methods	

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	2	3	3	1	–	–
CO-2	2	3	3	2	–	–
CO-3	2	3	3	1	–	–
CO-4	2	3	3	2	–	–
CO-5	2	3	3	2	–	–

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Introduction: Introduction to Reinforcement Learning (RL) – Difference between RL and Supervised Learning, RL and Unsupervised Learning. Elements of RL, Markov property, Markov chains, Markov reward process (MRP). Evaluative Feedback - Multi-Arm Bandit Problem: An n-Armed Bandit Problem, Exploration vs Exploitation principles, Action value methods, Incremental Implementation, tracking a non-stationary problem, optimistic initial values, upper-confidence-bound action selection, Gradient Bandits. Introduction to and proof of Bellman equations for MRPs. Introduction to Markov decision process (MDP), state and action value functions, Bellman expectation equations, optimality of value functions and policies, Bellman optimality equations. Dynamic Programming (DP): Overview of dynamic programming for MDP, principle of optimality, Policy Evaluation, Policy Improvement, policy iteration, value iteration, asynchronous DP, Generalized Policy Iteration. Monte Carlo Methods for Prediction and Control: Overview of Monte Carlo methods for model free RL, Monte Carlo Prediction, Monte Carlo estimation of action values, Monto Carlo Control, On policy and off policy learning, Importance sampling. Temporal Difference Methods: TD Prediction, Optimality of TD (0), TD Control methods - SARSA, QLearning and their variants.

Learning Resources:

Text Books:

1. Richard S. Sutton and Andrew G. Barto, Reinforcement Learning: An Introduction, MIT Press 2020/Bradford Books 2018, Second Edition.

Reference Books:

1. Csaba Szepesvari, Algorithms for Reinforcement Learning, Morgan & Claypool, 2010, First edition.

2. Warren B. Powell, Reinforcement Learning and Stochastic Optimization: A Unified Framework for Sequential Decisions, Wiley, 2022, First Edition.

CD18047 **3-0-0 (3)**

Social Media Analytics

Pre-Requisites: **None**

Course Outcomes:

CO-1	Understand the importance of social media and networks
CO-2	Enhance analytical skills for analyzing social media and networking data
CO-3	Develop skills to leverage extended enterprise data
CO-4	Create real-life case studies using social media data

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	-	1	-	1	-	1
CO-2	1	3	2	1	-	1
CO-3	1	2	2	1	1	1
CO-4	1	2	1	1	-	1

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

Introduction to social network analysis: Vertex or node, edge, neighbors, degree, shortest path, cycle, tree, complete graph, bipartite graphs, directed graphs, weighted graphs, adjacency matrix;

Social networks examples (Facebook, movie collaboration, and paper collaboration), information networks (web), biological networks (neural networks, ecological networks), random graphs with general degree distributions, models of network formation, Properties of Large-Scale Networks: Six- degree separation, scale-free distributions, small-world effect, and strong community structure;

Networks and Centrality Measures: Degree, closeness, betweenness, edge betweenness, eccentricity, clustering coefficient, eigenvector; Spread of influence through a network, influence maximization in networks, spread of disease on networks, Information networks;

Community Detection and graph based clustering: communities in social media, node-centric community detection, group-centric community detection, network-centric community detection, hierarchy-centric community detection, Topology discovery. Community Evaluation;

Link Prediction: Challenges in link prediction, link prediction methods and algorithms, clustering approaches for link prediction;

Sentiment Analysis: Sentiments and Opinions, lexicon based methods, machine learning based methods, feature-based sentiment analysis, slang sentiment analysis;

Social Listening and Social Recommendation Systems: Social Recommendation Using collaborative filtering, community detection and probabilistic matrix factorization

Learning Resources:

Text Books:

1. Reza Zafarani, Mohammad Ali Abbasi, Huan Liu, Social Media Mining – An Introduction, Cambridge University Press, 2014.
2. Charu C Aggarwal (Ed.), Socail Network Data Analytics, Springer, 2011.
3. Hansen, Derek, Ben Sheiderman, Marc Smith., Analyzing Social Media Networks with NodeXL: Insights from a Connected World, Morgan Kaufmann, 2011.

CD18049 **3-0-0 (3)**

User Interface Design

Pre-Requisites: **None**

Course Outcomes:

CO-1	Understand the characteristics of Graphical and Web User Interfaces
CO-2	Use the principles of the development process
CO-3	Apply the interaction styles
CO-4	Analyze the design issues

Course Articulation Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
CO-1	1	1	-	1	1	1
CO-2	1	1	-	2	2	2
CO-3	2	2	-	2	2	2
CO-4	2	2	-	2	2	2

1 - Slightly; **2 - Moderately;** **3 - Substantially**

Syllabus:

The Importance of the User Interface.

Usability of Interactive Systems: Guidelines, Principles, and Theories.

Characteristics of Graphical and Web User Interfaces: The Graphical User Interface, the Web User Interface.

Development Processes: Managing Design Processes, Evaluating Interface Designs, Software Tools.

Interaction Styles: Direct Manipulation and Virtual Environments, Menu Selection, Form Filling, and Dialog Boxes, Command and Natural Languages, Interaction Devices, Collaboration.

Design Issues: Quality of Service, Balancing Function and Fashion, User Manuals, Online Help, and Tutorials, Information Search and Visualization.

Learning Resources:

Text Books:

1. Ben Sheiderman, Design The User Interface, Pearson Education, 1998.
2. Wilbent. O. Galitz , The Essential Guide To User InterfaceDesign, John Wiley& Sons, 2001.