NATIONAL INSTITUTE OF TECHNOLOGY WARANGAL

SCHEME OF INSTRUCTION AND SYLLABI

Masters of Computer Applications

Effective from 2021-22

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

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 a globally competitive environment.
- Allowing stakeholders 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 societal needs.

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

VISION

Attaining global recognition in Computer Science & Engineering education, research and training to meet the growing needs of the industry and society

MISSION

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

Department of Computer Science and Engineering:

Brief about the Department:

The Department of Computer Science and Engineering was established in the year 1991. The department offers high quality undergraduate, postgraduate and doctoral programs. The B. Tech. (Computer Science and Engineering) program was started in the year 1983 with an intake of 20 students. The intake was subsequently increased to 120 in 2008. M. Tech (Computer Science and Engineering) program was started in 1987 with an intake of 18 and subsequently increased to 20 in 2008. M. Tech (Information Security) was introduced in the year 2008 Under ISEAP sanctioned by Ministry of Communication and Information Technology (MCIT), DOE, GOI, New Delhi with intake of 20. Later, it was renamed as Computer Science and Information Security. The Master of Computer Applications (MCA) program was started in 1986 with an intake of 30 and increased to 46 from 2008. B. Tech, M. Tech. (CSE) and M. Tech. (CSIS) programs were accredited in 2014 by NBA as per Washington Accord.

List of Programs offered by the Department:

Program	Title of the Program
B.Tech.	Computer Science and Engineering
M. Tech.	Computer Science and Engineering
	Computer Science and Information Security
MCA	Master in Computer Applications
Ph. D.	Computer Science and Engineering

Note: **Refer to the following weblink for Rules and Regulations of PG programs:**
https://nitw.ac.in/main/%20RulesandRegulations/PGProgrammes/

Masters of Computer Applications

Program Educational Objectives (PEOs)

PEO1	Design applications for real-world problems and analyze their Complexities by testing.
PEO2	Design and develop user interface frameworks among the subsystems to enhance portability.
PEO3	Discover knowledge from large data sets to analyze technical solutions for complex applications.
PEO4	Work in teams to learn and assess security, privacy, cost and quality assurance in developing software systems.
PEO5	Engage in lifelong learning to keep pace with changing landscape of technologies for professional advancement.

Program Articulation Matrix

Mission Statement	PEO1	PEO2	PEO3	PEO4	PEO5
Imparting quality education through a well-designed curriculum in tune with the challenging software needs of the industry	3	2	2	1	2
Providing state-of-art research facilities to generate knowledge and develop technologies in the thrust areas of computer science and engineering.	2	2	3	1	2
Developing linkages with world-class organizations to strengthen industry-academia relationships for mutual benefit.	2	2	2	3	1

1 - Slightly; 2 - Moderately; 3 – Substantially

Program Outcomes (POs)

At the end of the program, the student will be able to:

PO1	Apply mathematical foundation and domain knowledge for conceptualizing the computing models for real-life problems.
PO2	Design solutions for complex business scenarios or processes that meet the specific needs of societal problems.
PO3	Infer and predict the knowledge from data and provide synthesis to derive valid conclusions.
PO4	Create, select, and apply appropriate techniques, resources, modern computing tools, and skills required for innovative software solutions.
PO5	Ability to design, develop, deploy and manage robust and reliable software projects by satisfying the realistic, economic, social, safety and security constraints.
PO6	Demonstrate a higher level of professional skills to communicate with the peer fraternity by effectively presenting the reports on complex activities.

MAPPING OF PROGRAM OUTCOMES WITH PROGRAME EDUCATIONAL OBJECTIVES

PO	PEO1	PEO2	PEO3	PEO4	PEO5
1	2	3	2	1	1
2	3	3	2	2	2
3	2	1	3	2	2
4	2	2	2	2	2
5	3	2	2	3	2
6	1	2	2	1	2

1 - Slightly; 2 - Moderately; 3 – Substantially

CURRICULAR COMPONENTS

Degree Requirements for MCA

Category Description	Credits
Basic Science and Humanities (BSH)	9
Professional Core Courses (PCC)	78
Professional Elective Courses(PEC)	18
Seminar	1
Comprehensive Viva- Voce	2
Dissertation Work	12
Total	**120**

SCHEME OF INSTRUCTION

MCA-I Course Structure

I – Year: I – Semester

S.No.	Course Code	Course Title	L	T	P	Credits	Cat. Code
1	MA4037	Statistics and Queuing Theory	3	0	0	3	BSH
2	SM4331	Managerial Economics	3	0	0	3	BSH
3	CS4301	Mathematical Foundations of Computer Science	3	0	0	3	PCC
4	CS4302	Computer Organization	3	0	0	3	PCC
5	CS4303	Problem Solving and Programming	3	0	0	3	PCC
6	CS4304	Computer Game Development and Animation	0	1	2	2	PCC
7	CS4305	R Programming Lab	0	1	2	2	PCC
8	CS4306	Problem Solving and Programming Lab	0	0	3	1.5	PCC
		Total	**15**	**2**	**7**	**20.5**	

I – Year: II – Semester

S.No.	Course Code	Course Title	L	T	P	Credits	Cat. Code
1	CS4351	Database Systems	3	0	0	3	PCC
2	CS4352	Data Structures	3	0	0	3	PCC
3	CS4353	Web Technologies	3	0	0	3	PCC
4	CS4354	Object Oriented Programming	3	0	0	3	PCC
5	SM4381	Management Theory and Practice	3	0	0	3	BSH
6	CS4355	Database Systems Lab	0	0	2	1	PCC
7	CS4356	Data Structures Lab	0	0	3	1.5	PCC
8	CS4357	Web Technologies Lab	0	0	3	1.5	PCC
9	CS4358	Object Oriented Programming Lab	0	0	3	1.5	PCC
		Total	**15**	**0**	**11**	**20.5**	

II– Year: I – Semester

S.No.	Course Code	Course Title	L	T	P	Credits	Cat. Code
1	CS5301	Advanced Data Structures	2	0	0	2	PCC
2	CS5302	Operating System Concepts	3	0	0	3	PCC
3	CS5303	Software Engineering Principles	3	0	0	3	PCC
4	CS5304	Computer Communications and Network	3	0	0	3	PCC
5	CS5305	Advanced Databases	3	0	0	3	PCC
6		Elective - 1	3	0	0	3	PEC
7	CS5306	Operating System Concepts Lab	0	0	2	1	PCC
8	CS5307	Software Engineering Lab	0	0	2	1	PCC
9	CS5308	Computer Communications and Network Lab	0	0	2	1	PCC
10	CS5309	Advanced Databases Lab	0	0	2	1	PCC
		Total	**17**	**0**	**8**	**21**	

II– Year: II – Semester

S. No.	Course Code	Course Title	L	T	P	Credits	Cat. Code
1	CS5351	Mobile Applications Development	1	0	3	2.5	PCC
2	CS5352	Software Testing	3	0	0	3	PCC
3	CS5353	Algorithm Analysis and Design	3	0	0	3	PCC
4	CS5354	Principles of Data Warehousing and Data Mining	3	0	0	3	PCC
5	CS5355	Python Programming	2	0	2	3	PCC
6		Elective – 2	3	0	0	3	PEC
7		Elective – 3	3	0	0	3	PEC
8	CS5356	Knowledge Engineering Lab	0	0	2	1	PCC
9	CS5357	Software Testing Lab	0	0	2	1	PCC
		Total	**18**	**0**	**9**	**22.5**	

III – Year: I – Semester

S. No.	Course Code	Course Title	L	T	P	Credits	Cat. Code
1	CS6301	Cryptography and Network Security	3	0	0	3	PCC
2	CS6302	Internet of Things	3	0	0	3	PCC
3	CS6303	IoT and Network Security Lab	0	0	3	1.5	PCC
4	CS6304	Machine Learning	3	0	0	3	PCC
5		Elective – 4	3	0	0	3	PEC
6		Elective – 5	3	0	0	3	PEC
7		Elective – 6	3	0	0	3	PEC
8	CS6305	Machine Learning lab	0	0	2	1	PCC
9	CS6347	Comprehensive Viva-voce	0	0	4	2	CVV
10	CS6348	Seminar	0	0	2	1	SEM
		Total	**18**	**0**	**11**	**23.5**	

III – Year: II – Semester

S. No.	Course Code	Course Title	L	T	P	Credits	Cat. Code
1	CS6399	Dissertation Work	0	0	0	12	DW
		Total				**12**	

Note: PCC – Professional Core Courses
PEC – Professional Elective Courses
BSH – Humanities and Social Science Core

Credit Distribution Table – Semester-wise and Category-wise

Cat. Code	Credits in Each Semester						
	Semester I	Semester II	Semester III	Semester IV	Semester V	Semester VI	Total
BSH	6	3	0	0	0	0	9
PCC	14.5	17.5	18	16.5	11.5	0	78
PEC	0	0	3	6	9	0	18
SEM	0	0	0	0	1	0	1
CVV	0	0	0	0	2	0	2
DW	0	0	0	0	0	12	12
Total	20.5	20.5	21	22.5	23.5	12	120

Departmental Elective Courses

Electives for II Year I Sem

S.No.	Course Code	Course Title	L	T	P	Credits	Cat. Code
1	CS5311	Artificial Intelligence	3	0	0	3	PEC
2	CS5312	Information Systems Management	3	0	0	3	PEC
3	CS5313	Information Security	3	0	0	3	PEC
4	CS5314	Modeling and Simulation	3	0	0	3	PEC
5	CS5315	Advanced Web Technologies	3	0	0	3	PEC

Electives for II Year II Sem

S. No.	Course Code	Course Title	L	T	P	Credits	Cat. Code
1	CS5361	Cloud Computing	3	0	0	3	PEC
2	CS5362	Distributed Operating Systems	3	0	0	3	PEC
3	CS5363	Ubiquitous Computing	3	0	0	3	PEC
4	CS5364	Image Processing	3	0	0	3	PEC
5	CS5365	Parallel Computing	3	0	0	3	PEC
6	CS5366	E-Commerce Technologies and Management	3	0	0	3	PEC
7	CS5367	Network Programming	3	0	0	3	PEC
8	CS5368	GPU Programming	3	0	0	3	PEC
9	CS5369	Software Architecture	3	0	0	3	PEC
10	CS5370	Randomized Algorithms	3	0	0	3	PEC
11	CS5371	User Interface Design	3	0	0	3	PEC

Electives for III Year I Sem

S.No.	Course Code	Course Title	L	T	P	Credits	Cat. Code
1	CS6311	Digital Forensics	3	0	0	3	PEC
2	CS6312	Human Computer Interaction	3	0	0	3	PEC
3	CS6313	Design Patterns	3	0	0	3	PEC
4	CS6314	Big Data Technologies	3	0	0	3	PEC
5	CS6315	Bioinformatics	3	0	0	3	PEC
6	CS6316	ERP and Supply Chain Management	3	0	0	3	PEC
7	CS6317	Web Analytics	3	0	0	3	PEC
8	CS6318	Algorithmic Graph Theory	3	0	0	3	PEC
9	CS6319	Program Analysis and Verification	3	0	0	3	PEC
10	CS6320	Foundations of Data Science	3	0	0	3	PEC
11	CS6321	Foundations of Blockchain	3	0	0	3	PEC
12	CS6322	Game Theory	3	0	0	3	PEC
13	CS6323	Quantum Computing	3	0	0	3	PEC
14	CS6324	Social Media Analytics	3	0	0	3	PEC
15	CS6325	Malware Detection and Mitigation	3	0	0	3	PEC
16	CS6326	Deep Learning	3	0	0	3	PEC
17	CS6327	Mobile Computing	3	0	0	3	PEC
18	CS6328	Soft Computing Techniques	3	0	0	3	PEC

DETAILED SYLLABUS

Course Code: MA4037	Statistics and Queueing Theory	Credits 3-0-0: 3

Pre-Requisites: Nil

Course Outcomes:

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

CO1	Make use of the concept of random variables to solve problems arising in probability distributions
CO2	Explain the concepts of estimates
CO3	Perform hypothesis tests on small and large samples
CO4	Find the coefficient of correlation and lines of regression
CO5	Understand the characteristics of a queueing model

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	-	3	1	-	2
CO2	3	-	3	-	-	2
CO3	3	-	3	-	-	2
CO4	3	-	3	-	-	2
CO5	3	-	-	2	-	-

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

Text Books/ Reference Books/ Online 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:

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

Course Code: SM - 4331	Managerial Economics	Credits 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Understand the domain knowledge in micro economic concepts and apply it to business decisions
CO2	Conceptualize the economic environment of business.
CO3	Understand the Social responsibility of business.
CO4	Identify areas of social problems and solve the same with ITES.
CO5	Develop effective presentation skills with special emphasis on contemporary topics.

Course Articulation Matrix:

PO / PO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	-	-	-	-	1
CO2	3	-	-	-	-	1
CO3	-	3	-	-	-	1
CO4	-	-	2	1	1	1
CO5	-	-	-	-	1	3

Syllabus:

Chapter-1- Introduction to Managerial economics - Introduction to Economics and Managerial Economics: Managerial economics as decision sciences and bridging the gap between theory and practice. Fundamental concepts used in business decisions

Chapter-2 - Fundamentals of Micro Economics - Analysis of Consumer Demand, Analysis of market demand and demand elasticities, Methods of Demand Forecasting. Theory of cost and Break-Even analysis, Market structure and pricing decisions - Perfect competition, Monopoly, Oligopoly and Monopolistic competition.

Chapter- 3 - Macro Economics- Introduction to Macro Economics and National Income –Circular flow Model of the Economy, National Income, Concepts, measurement, theory of National income determination, Income determination with Government and Foreign trade

Chapter- 4 - Inflation- Introduction to the concept of Inflation - meaning, theories and control measures by the Government

Chapter- 5 - Business Cycles and stabilization- Business/ trade cycles- meaning, Strategies to cope up – Stabilization by the government.

Chapter- 6 - Balance of Payments – Meaning - Exchange Rate and its Determination -

Chapter- 7 - Economic policy of 1991 - Features, Liberalization, Globalization and Privatization, Impact on IT Industry and ITES. Present Policies of the Government.

Chapter- 8 – Welfare Economics **–** Social Responsibility of Private Business – Introduction – social responsibility of business and businessman – Indian scenario.

Text Books/ Reference Books/ Online Resources:

Text Books-

1. "Managerial Economics", D.N. Dwivedi, Vikas publishing house, Eighth Edition, 2015
2. "Managerial Economics" Geetika, Piyali Ghosh and Purba Roy Choudhury, Mc Graw Hill Education, Third edition. 2018.

Readings-

1. Managerial Economics and Strategy, Jeffrey M Perloff, James A Brander, Pearson, Third Edition.
2. Managerial Economics, Matthew Metzgar, Legacy publishers,
3. Managerial Economics and Business Strategy, Michael R Baye and Jeffrey T Prince, 9th edition, e-book only.

Course Code: CS4301	Mathematical Foundations of Computer Science	Credits 3-0-0: 3

Pre-Requisites: Nil

Course Outcomes:

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

CO1	Apply formal methods of proof to solve discrete problems
CO2	Apply Propositional logic and First order logic to solve problems
CO3	Apply techniques for counting the occurrences of discrete events includingpermutations, combinations with or without repetitions
CO4	Formulate and solve graph problems including searching and spanning
CO5	Formulate and solve recurrence relations

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	2	1	2	1	1
CO2	3	2	1	1	1	1
CO3	3	3	-	1	1	1
CO4	3	3	2	2	1	1
CO5	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

Text Books/ Reference Books/ Online Resources:

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.

Course Code: CS4302	Computer Organization	Credits 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Fundamentals of digital computers, and internal organization of computers
CO2	Analysis of the CPU, memory unit, Input/Outputs and the relations between its main components.
CO3	Analysis of the memory hierarchy and perform computations with the functional units of the processor and memory.
CO4	Cost performance and design trade-offs in designing and constructing a computer processor including memory.
CO5	Apply the knowledge of combinational and sequential logical circuits to design computer architecture to calculate the arithmetic expressions.

Course Articulation Matrix:

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	1	1	1	1	1	-
CO2	2	1	2	2	1	1
CO3	3	1	1	2	2	1
CO4	3	2	1	2	2	1
CO5	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

Text Books/ Reference Books/ Online Resources:

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

Course Code: CS4303	Problem Solving and Programming	Credits 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Design algorithms for solving simple mathematical problems including computing,searching and sorting
CO2	Compare and contrast algorithms in terms of space and time complexity to solve simple mathematical problems
CO3	Explore the internals of computing systems to suitably develop efficient algorithms
CO4	Examine the suitability of data types and structures to solve specific problems
CO5	Apply control structures to develop modular programs to solve mathematical problems
CO6	Apply object-oriented features in developing programs to solve real-world problems

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	1	2	2	2	1
CO2	2	1	2	2	2	1
CO3	1	2	3	1	3	1
CO4	1	2	1	2	2	1
CO5	3	3	1	2	2	2
CO6	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.

Text Books/ Reference Books/ Online Resources:

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

Course Code: CS4304	Computer Game Development and Animation	Credits 0-1-2: 2

Course Outcomes:

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

CO1	Implement game functionality through scripting
CO2	Implement 2D/3D Graphics, Animation
CO3	Apply the physics and character control needed in a game.
CO4	Design a game based on the concepts discussed in this course.

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2	1	-	1	2	-
CO2	2	1	-	2	1	-
CO3	2	2	1	2	2	1
CO4	3	3	2	2	3	1

1 - Slightly; 2 - Moderately; 3 – Substantially

Syllabus:

Introduction: Types of games, Design and Development processes, Computer gaming, Game Engines.

Working with Unity: working with Game objects, Components, Prefabs, Scenes and Asset; building Unity Projects.

Scripting: Mono Behaviour, Call backs, creating frame rate-independent behaviour, accessing Components, finding objects. using coroutines, loading a Level, storing data on disk, saving and loading the Game State, storing data in Assets using Scriptable Object.

Input: working with keyboard and mouse input, customizing Unity's input system.

Math: storing coordinates of varying dimensions using Vectors, rotating in 3D space, transformations in 3D space with matrices, working with Angles.

2D Graphics: importing Sprites, adding a Sprite to the Scene, creating a Sprite animation, creating a Sprite with 2D physics, customizing Sprite collision shapes, applying forces to 2D objects, using a custom Material for Sprites, Sprite sorting, creating a 2.5D Scene.

3D Graphics: creating a simple Material, controlling a Material's Property through a Script, setting up a Material using Textures, making a Material use a Shader, animating a Shader over time, controlling the Speed of an animated Shader, using Baked Lighting and Real-Time Lighting, using Baked Emission Sources, making Static Objects Cast Shadows on Dynamic Objects, using Light Probes to Influence Lighting, using Reflection Probes, faking a Dynamic Emissive Object, loading Textures from Disk, rendering to a Texture.

Physics and Character Control: understanding FixedUpdate, implementing Mouselook, controlling a 3D Character, interacting with Switches and Objects, picking Up and Putting Down Objects, detecting when an Object is Touching another Object, detecting when an Object is in a Trigger Area.

Animation and Movement: animating an Object, basic Character Movement, Inverse Kinematics, Navigation and Animating in Sync, Cinematic Camera Tracking, automatically Switching Cameras, keeping Multiple Objects in View.

Gameplay: managing Quests and Hitpoints, creating a Top-Down Camera, dragging a Box to Select Objects, creating a Menu Structure, creating a Wheeled Vehicle, keeping a Car from Tipping Over, creating Speed Boosts, creating a Camera that Orbits around its Target, creating Orbiting Cameras that won't Clip through Walls, detecting when the Player has Completed a Lap.

Behaviour and AI: defining a Path that AI Entities and the Player can follow, letting Entities in your Game follow a Path, enemies Detecting when they can see the Player, enemies Detecting where they can Take Cover, building and using a State Machine.

Sound and Music: Playing Sounds, using Audio Effects.

User Interface: Working with UI Controls, creating a List of Items, Custom Editors, Asset Processing.

Laboratory:

Lab programs/procedures to understand and demonstrate:

1. Introduction to Unity
2. Scripting in C#
3. 2D & 3D Graphics
4. Animation
5. Gameplay
6. Behaviour and AI
7. Creating a full game

Text Books / Reference Books / Online Resources:

1. Paris Buttfield-Addison, Jon Manning and Tim Nugent, "*Unity Game Development Cookbook*", 1st Edition, O'Reilly Media Inc., 2019.
2. Jeremy Gibson Bond, *"Introduction to Game Design, Prototyping, and Development"*, 3rd Edition, Addison-Wesley Professional, 2021.

Course Code: CS4305	R Programming Lab	Credits 0-1-2: 2

Pre-requisites: Nil

Course Outcomes:

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

CO1	Understand the fundamental concepts of R programming
CO2	Develop programs based on iterations and control statements
CO3	Analyse the various data formats and their representations
CO4	Execute, verify and validate queries for specific domains
CO5	Generate the various data visualizations

Course Articulation Matrix:

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2	2	1	2	1	1
CO2	2	2	1	2	3	1
CO3	2	2	2	2	2	2
CO4	1	3	2	2	2	1
CO5	1	1	1	1	1	3

1 - Slightly; 2 - Moderately; 3 – Substantially

Syllabus:

Installing R and RStudio, RStudio Overview, Working in the Console, Arithmetic Operators, Logical Operations, Using Functions, Getting Help in R and Quitting RStudio

Creating Variables, Numeric, Character and Logical Data, Vectors, Data Frames, Factors Sorting Numeric, Character, and Factor, Vectors, Special Values

Packages:

Installing and loading packages, Downloading and importing data, Writing R scripts, Creating reports, Adding comments and documentation

Statistics in R

Measures of variability, Skewness and kurtosis, Summary functions, Correlations

Graphs:

Box Plots, Scatter Plots, Histograms

Iteration and conditional statements:

while loops, for loops, If / else Boolean logical operators

Data querying:

Writing SQL statements in R, Using the Select, From, Where, Is, Like, Order By, Limit, Max, Min SQL functions

Text Books/ Reference Books/ Online Resources:

1. R Project: http://www.r-project.org/
2. RStudio (additional libraries required): http://www.rstudio.com
3. Quick-R http://www.statmethods.net/
4. Google's R Style Guide: http://google-styleguide.googlecode.com/svn/trunk/ Rguide.xml

Course Code: CS4306	Problem Solving and Programming Lab	Credits 0-0-3: 1.5

Pre-requisites: Nil

Course Outcomes:

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

CO1	Design and test programs to solve mathematical and scientific problems
CO2	Design and test complex control structures for real-life problems
CO3	Implement modularity for complex, reliable and robust structures
CO4	Design solutions for the real-life problems using OOP paradigm

Course Articulation Matrix:

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	2	2	1	2	1
CO2	1	1	2	1	2	2
CO3	1	2	3	2	2	1
CO4	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

Text Books/ Reference Books/ Online Resources:

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.

Course Code: CS4351	Database Systems	Credits 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Understand functional components of the DBMS.
CO2	Devise queries using Relational Algebra, Relational Calculus and SQL.
CO3	Design database schema.
CO4	Develop E-R model.
CO5	Evaluate and optimize queries.
CO6	Analyze transaction processing, concurrency control and recovery techniques.

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2	-	1	-	1	1
CO2	2	2	2	2	2	1
CO3	2	3	2	2	3	1
CO4	2	2	1	1	1	2
CO5	2	1	1	1	3	1
CO6	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.

Text Books/ Reference Books/ Online Resources:

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.

Course Code: CS4352	Data Structures	Credits 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Analysis of algorithms and fundamental concepts of ADT, identification of suitable data structures to solve complex problems
CO2	Fundamentals of Arrays, Stack and Queues, develop and analysis of the algorithms for different operations.
CO3	Fundamentals of Binary Trees and Graphs, develop algorithms for various operations performed on trees and graphs.
CO4	Implementation and analysis of sorting and searching techniques
CO5	Implement symbol table using hashing techniques and multi-way search trees

Course Articulation Matrix:

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	1	2	-	1	1	2
CO2	2	1	1	2	2	1
CO3	2	3	1	1	2	2
CO4	2	2	2	3	2	2
CO5	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

Text Books/ Reference Books/ Online Resources:

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

Course Code: CS4353	Web Technologies	Credits 3-0-0:3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Understand, analyze and build dynamic and interactive websites
CO2	Understand current and evolving Web languages for integrating media and userinteraction in both front end and back end elements of a Web site
CO3	Analysis and reporting of web data using web analytics
CO4	Applying different testing and debugging techniques and analyzing the website effectiveness.

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2	1	-	2	3	1
CO2	-	1	-	2	3	3
CO3	-	1	2	2	3	1
CO4	-	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.

Text Books/ Reference Books/ Online Resources:

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

Course Code: CS4354	Object Oriented Programming	Credits 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Understand fundamental concepts in an object-oriented approach.
CO2	Analyze design issues in developing OOP applications.
CO3	Write computer programs to solve real-world problems in Java.
CO4	Analyze source code API documentations.
CO5	Create GUI-based applications.

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	1	1	-	1	1	-
CO2	1	3	-	2	3	2
CO3	2	2	2	2	2	2
CO4	1	2	-	3	2	3
CO5	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

Text Books/ Reference Books/ Online Resources:

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

Course Code: SM4381	Management Theory and Practice	Credits: 3-0-0: 3

Course Outcomes**:**

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

CO1	Apply management theories, models and principles to practice through analysis of case studies and contemporary business events.
CO2	Outline the historical evolution of management theories.
CO3	Describe the four management functions of planning, organizing, leading, and controlling
CO4	Understand different organizational functions and business processes

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	-	2	-	-	-	3
CO2	-	2	-	-	1	3
CO3	-	2	-	-	1	3
CO4	-	2	-	-	1	3

Syllabus

Chapter 1- 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.

Chapter 2 – Planning: Nature and purpose of planning, planning process, types of planning, Objectives, Planning premises, Planning Tools and Techniques, Decision making steps and process.

Chapter 3 – 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.

Chapter 4 - 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.

Chapter 5 – Controlling: System and process of controlling, budgetary and non-budgetary control techniques, Productivity problems and management, control and performance, direct and preventive control – reporting.

Text Books/ Reference Books/ Online Resources:

Text Books:

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

Course Code: CS4355	Database Systems Lab	Credits 0-0-2: 1

Pre-requisites: Nil

Course Outcomes:

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

CO1	Design and Implement a database schema
CO2	Devise queries using DDL, DML, DCL and TCL commands.
CO3	Develop application programs using PL/SQL
CO4	Design and implement a project using embedded SQL and GUI.
CO5	Apply modified components for performance tuning in open source software.

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2	2	2	2	2	2
CO2	2	1	2	2	2	2
CO3	2	2	2	2	2	2
CO4	2	2	2	2	2	1
CO5	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;

Text Books/ Reference Books/ Online Resources:

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.

Course Code: CS4356	Data Structures Lab	Credits 0-0-3: 1.5

Pre-requisites: Nil

Course Outcomes:

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

CO1	Develop ADT for stack and queue applications
CO2	Implement tree and graph algorithms
CO3	Implement and analyze internal and external sorting algorithms
CO4	Design and implement symbol table using hashing technique

Course Articulation Matrix:

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2	1	-	2	2	1
CO2	2	1	1	2	2	1
CO3	3	2	1	3	1	1
CO4	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

Text Books/ Reference Books/ Online Resources:

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

Course Code: CS4357	Web Technologies Lab	Credits 0-0-3: 1.5

Pre-requisites: Nil

Course Outcomes:

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

CO1	Understand, analyze and build dynamic and interactive websites
CO2	Understand current and evolving Web languages for integrating media and userinteraction in both front end and back end elements of a Web site
CO3	Analysis and reporting of web data using web analytics
CO4	Applying different testing and debugging techniques and analyzing the website effectiveness.

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2	1	-	2	3	1
CO2	-	1	-	2	3	3
CO3	-	1	2	2	3	1
CO4	-	1	-	2	3	2

1 - Slightly; 2 - Moderately; 3 – Substantially

Syllabus:

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

Text Books/ Reference Books/ Online Resources:

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.

Course Code: CS4358	Object Oriented Programming Lab	Credits 0-0-3: 1.5

Pre-requisites: Nil

Course Outcomes:

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

CO1	Understand fundamental concepts in an object-oriented approach.
CO2	Analyze design issues in developing OOP applications.
CO3	Write computer programs to solve real-world problems in Java.
CO4	Analyze source code API documentations.
CO5	Create GUI-based applications.

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	1	1	-	1	1	-
CO2	1	3	-	2	3	2
CO3	2	2	2	2	2	2
CO4	1	2	-	3	2	3
CO5	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

Text Books/ Reference Books/ Online Resources:

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

Course Code: CS5301	Advanced Data Structures	Credits: 2-0-0: 2

Pre-requisites: CS4352 - Data Structures

Course Outcomes:

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

CO1	Understand the implementation of symbol table using hashing techniques.
CO2	Develop and analyze algorithms for different types of trees, heaps, skip lists and tries.
CO3	Develop algorithms for text processing applications
CO4	Identity suitable data structures and develop algorithms for computational geometry problems

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	1	2	-	2	3	-
CO2	2	3	-	2	3	-
CO3	2	3	-	2	2	-
CO4	2	3	-	2	2	-

1 - Slightly; 2 - Moderately; 3 – Substantially

Syllabus:

Review of Hashing, Dynamic Hashing, cuckoo hashing, Need for Randomizing Data Structures and Algorithms, Search and Update Operations on Skip Lists, Probabilistic Analysis of Skip Lists, Red Black Trees, B- Trees, B+-Trees, Splay Trees, Digital Search Trees, Finger Search Tree, d-Heaps, Leftist Heaps, Skew Heaps, Binomial Heaps, Fibonacci Heaps, Brute-Force Pattern Matching, The Boyer-Moore Algorithm, The Knuth-Morris-Pratt Algorithm, Standard Tries, Compressed Tries, Suffix Tries, One Dimensional Range Searching, Two Dimensional Range Searching, Constructing a Priority Search Tree, Searching a Priority Search Tree, Priority Range Trees, Quad trees, k-D Trees.

Text Books/ Reference Books/ Online Resources:

1. Mark Allen Weiss, *Data Structures and Algorithm Analysis in C++*, 4th Edition, Pearson, 2004.
2. Michael T Goodrich and Roberto Tamassia, *Algorithm Design and Applications*, John Wiley, 2014.
3. Michael T Goodrich, Roberto Tamassia and David Mount, *Data Structures and Algorithms in C++*, Second Edition John Wiley & Sons, Inc., 2016.
4. Ellis Horowitz, Dinesh Mehta, Sartaj Sahni, *Fundamentals of Data Structures in C++*, University Press, 2008.

Course Code: CS5302	Operating System Concepts	Credits 3-0-0: 3

Pre-requisites: CS4302-Computer Organization

Course Outcomes:

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

CO1	Distinguish functional architectures of operating systems and file systems
CO2	Develop algorithms for subsystem components
CO3	Design device drivers and multi-threading libraries for an OS
CO4	Develop application programs using UNIX system calls
CO5	Design and solve synchronization problems

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	1	1	-	1	1	1
CO2	2	2	-	2	2	1
CO3	2	2	-	1	2	2
CO4	2	2	-	2	2	1
CO5	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.

Text Books/ Reference Books / Online Resources:

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

Course Code: CS5303	Software Engineering Principles	Credits: 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Comprehend software development life cycle
CO2	Prepare SRS document for a project
CO3	Apply software design and development techniques
CO4	Identify verification and validation methods in a software engineering project
CO5	Implement testing methods for software
CO6	Analyze and Apply project management techniques for a case study

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	1	3	1	2	2	3
CO2	-	1	1	1	3	2
CO3	1	3	-	1	3	2
CO4	1	-	2	2	1	3
CO5	2	1	-	3	3	2
CO6	1	-	2	3	3	3

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.

Text Books/ Reference Books / Online Resources:

1. Stephen R Schach, Object Oriented and Classical Software Engineering, 5/e, TMH, 2010

2. Ian Sommerville, Software Engineering, 9/e, Pearson, 2010.

Course Code: CS5304	Computer Communications and Network	Credits: 3-0-0: 3

Pre-requisites: CS5302: Operating System Concepts

Course Outcomes:

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

CO1	Understand OSI and TCP/IP models
CO2	Analyze MAC layer protocols and LAN technologies
CO3	Design applications using internet protocols
CO4	Implement routing and congestion control algorithms
CO5	Develop application layer protocols

Course Articulation Matrix:

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	-	1	-	-	-	-
CO2	-	1	1	-	-	-
CO3	3	3	-	3	-	-
CO4	1	2	1	-	-	-
CO5	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.

Text Books/ Reference Books / Online Resources:

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

Course Code: CS5305	Advanced Databases	Credits: 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Design distributed database for application development.
CO2	Apply query optimization principles for optimizing query performance in centralized and distributed database systems
CO3	Design distributed database schema using principles of fragmentation and allocation.
CO4	Apply distributed transaction principles for handling transactions in distributed database applications.
CO5	Apply distributed database administration principles for managing distributed database.

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2	2	1	2	2	1
CO2	3	1	1	2	2	1
CO3	2	2	1	1	2	1
CO4	2	1	1	2	2	1
CO5	2	2	1	2	2	1

1 - Slightly; 2 - Moderately; 3 – Substantially

Syllabus:

Distributed Databases: Introduction to Distributed Database Systems, Distributed Database System Architecture; Top-Down Approach, Distributed Database Design Issues, Fragmentation, Allocation, Database Integration, Bottom-up approach, Schema Matching, Schema Integration, Schema Mapping; Data and Access Control, View Management, Data Security; Query processing problem, Objectives of Query processing, Complexity of Relational Algebra Operations, Characterization of Query Processors, Layers of Query Processing; Query Decomposition, Normalization, Analysis, Elimination of Redundancy and Rewriting; Localization of Distributed Data, Reduction for primary Horizontal, Vertical, derived Fragmentation; Distributed Query Execution, Query Optimization, Join Ordering, Static& Dynamic Approach, Semi-joins, Hybrid Approach; Taxonomy of Concurrency control Mechanisms, Lock-Based Concurrency Control, Timestamp-Based Concurrency Control, Optimistic Concurrency Control, Deadlock Management; Heterogeneity issues Advanced Transaction Models, Distributed systems 2PC& 3PC protocols, Replication protocols, Replication and Failures, HotSpares; Parallel Databases: Introduction to Parallel Databases, Parallel Database System Architectures, Parallel Data Placement, Full Partitioning; Parallel Query Processing, Query Parallelism; Parallel Query Optimization, Search Space, Cost Model, Search Strategy; Load Balancing.

Text Books/ Reference Books / Online Resources:

1. M T Ozsu, Patrick Valduriez, *Principles of Distributed Database Systems*, Prentice Hall, 1999.
2. S. Ceri and G. Pelaggati, *Distributed Database System Principles and Systems*, MGH, 1985.

Course Code: CS5306	Operating System Concepts Lab	Credits: 0-0-2: 1

Pre-requisites: Nil

Course Outcomes:

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

CO1	Implement elementary UNIX system commands
CO2	Develop programs to test synchronization problems
CO3	Design and develop user level thread library
CO4	Design and implement file system

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	1	1	-	1	2	1
CO2	2	3	-	2	1	1
CO3	2	2	-	2	2	1
CO4	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

Text Books/ Reference Books / Online Resources:

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

Course Code: CS5307	Software Engineering Lab	Credits 0-0-2: 1

Pre-requisites: Nil

Course Outcomes:

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

CO1	Prepare Software Requirement Specification document
CO2	Prepare design document and compute effort estimates for a software project
CO3	Design UML diagram for a case study
CO4	Design and Develop test cases for a software

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	1	2	1	2	3	3
CO2	1	3	1	2	3	-
CO3	2	2	-	3	2	2
CO4	2	1	2	3	2	1

1 - Slightly; 2 - Moderately; 3 – Substantially

Syllabus:

Problem Analysis and Project Planning – Software Requirement Analysis – Modeling – Software Developments and Debugging – Software Testing

Text Books/ Reference Books / Online Resources:

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

Course Code: CS5308	Computer Communications and Network Lab	Credits 0-0-2: 1

Pre-requisites: CS5306: Operating System Concepts Lab

Course Outcomes:

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

CO1	Develop programs for client-server applications
CO2	Perform packet sniffing and analyze packets in network traffic.
CO3	Implement error detecting and correcting codes

Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	3	-	2	-	-
CO2	2	2	2	1	1	-
CO3	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.

Text Books/ Reference Books / Online Resources:

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

Course Code: CS5309	Advanced Databases Lab	Credits: 0-0-2: 1

Pre-requisites: Nil

Course Outcomes:

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

CO1	Ability to design and implement distributed databases
CO2	Implement concurrency and query processing mechanisms
CO3	Design and implement parallel databases
CO4	Design and implement schema mapping, schema matching and a fragmentation mechanisms

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	1	2	2	1	1	1
CO2	1	1	1	1	1	1
CO3	1	3	1	-	2	1
CO4	1	2	2	-	2	2

1 - Slightly; 2 - Moderately; 3 – Substantially

Syllabus:

1. Creative distributed databases
2. Design a distributed database for online book store.
3. Program for Schema Matching
4. Program on Schema Mapping
5. Design of Query Processors
6. Programs on concurrent and Transactions
7. Program on indexing and pre-processing
8. Designing of parallel databases
9. Programs for Horizontal and vertical fragmentation
10. Implementing concurrency control mechanism

Text Books/ Reference Books/ Online Resources:

1. M. Stonebraker, *Readings in Database Systems*, 2/e, Morgan Kauffman, 1993.
2. M T Ozsu, Patrick Valduriez, *Principles of DistributedDatabase Systems*, Prentice Hall, 1999
3. S. Ceri and G. Pelaggati, *Distributed Database SystemPrinciples and Systems*, MGH, 1985.

Course Code: CS5351	Mobile Applications Development	Credits 1-0-3: 2.5

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

CO1	Demonstrate their understanding of the fundamentals of Android operating systems
CO2	Demonstrate their skills of using Android software development tools
CO3	Demonstrate their ability to develop software with reasonable complexity on mobile platform
CO4	Demonstrate their ability to deploy software to mobile devices
CO5	Demonstrate their ability to debug programs running on mobile devices

Course Articulation Matrix

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	1	2	1	2	3	2
CO2	1	2	1	2	3	2
CO3	1	2	1	2	3	2
CO4	1	2	1	2	3	2
CO5	1	2	1	2	3	2

1 - Slightly; 2 - Moderately; 3 – Substantially

Syllabus:

Characteristics of mobile applications, History of mobile application frameworks, Overview of the Android framework, Application models of mobile application frameworks, User-interface design for mobile applications, Managing application data, Integrating with cloud services, Integrating networking, the OS and hardware into mobile, applications, Addressing enterprise requirements in mobile applications: performance, scalability, modifiability, availability and security, Testing methodologies for mobile applications, Publishing, deployment, maintenance, and management

Text Books / Reference Books / Online Resources:

1. Valentino Lee, Heather Schneider, and Robbie Schell, *Mobile Applications: Architecture, Design, and Development*, Prentice Hall,2004
2. Brian Fling, *Mobile Design and Development, O'ReillyMedia, 2009*.
3. Maximiliano Firtman, "*Programming the Mobile Web*", O'Reilly Media, 2010.
4. Christian Crumlish and Erin Malone, *Designing Social Interfaces*, O'Reilly Media, 2009. EdBurnett, *Hello, Android. Introducing Google's Mobile Development Platform*, Fourth Edition, Pragmatic Bookshelf, 2015.

Course Code: CS5352	Software Testing	Credits 3-0-0: 3

Pre-Requisites: Nil

Course Outcomes:

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

CO1	Analyze Various test processes and continuous quality improvement
CO2	Analyze Types of errors and fault models
CO3	Modeling the behavior using FSM
CO4	Application of software testing techniques in commercial environments

Course Articulation Matrix:

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2	-	-	2	3	-
CO2	1	-	1	1	2	1
CO3	1	1	-	1	2	-
CO4	1	3	-	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

Text Books / Reference Books / Online Resources:

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

Course Code: CS5353	Algorithm Analysis and Design	Credits 3-0-0: 3

Pre-requisites: CS4303: Problem Solving and Programming, CS4352: Data Structures

Course Outcomes:

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

CO1	Analyse time and space complexity
CO2	Identify algorithm design methodology to solve problems
CO3	Design algorithms for network flows
CO4	Distinguish between P and NP classes of problems
CO5	Analyse amortized time complexity

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2	2	-	1	3	1
CO2	2	2	1	2	2	1
CO3	2	3	-	2	2	1
CO4	2	1	-	-	3	1
CO5	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.

Text Books/ Reference Books/ Online Resources:

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

Course Code: CS5354	Principles of Data Warehousing and Data Mining	Credits 3-0-0: 3

Pre-requisites: CS4351: Database Systems.

Course Outcomes:

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

CO1	Understand stages in building a Data Warehouse
CO2	Apply preprocessing techniques for data cleansing
CO3	Analyze multi-dimensional modeling techniques
CO4	Analyze and evaluate the performance of algorithms for Association Rules.
CO5	Analyze Classification and Clustering algorithms

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2	1	2	1	1	1
CO2	2	2	2	1	1	1
CO3	2	2	3	2	2	1
CO4	2	1	3	2	2	1
CO5	2	1	3	2	2	1

1 - Slightly; 2 - Moderately; 3 – Substantially

Syllabus:

Data Warehouse vs Databases – KDD Process - Data preprocessing Data Warehousing and OLAP technologies – Multi-Dimensional modeling, data warehouse architecture – Data Mining techniques – Association, Classification, Clustering, Sequential Patterns.

Text Books:

1. Jiawei Han and Kamber, M, *Data Mining Concepts andTechniques*, 2/e, Elsevier Publications, 2006.
2. Vipin Kumar, Michael Steinbach, *Introduction to Data Mining*,1/e, Addition-Wesley, 2006.

Course Code: CS5355	Python Programming	Credits 2-0-2: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Understand the fundamental concepts of Python programming
CO2	Develop programs based on iterations and control statements
CO3	Analyse the various data formats and their representations
CO4	Use various Scientific computing tools and databases
CO5	Generate the various data visualizations

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2	2	1	2	1	1
CO2	2	2	1	2	3	1
CO3	2	2	2	2	2	2
CO4	1	3	2	2	2	1
CO5	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, File handling, Object Oriented Programming in Python, Exception Handling in Python, Different modules and packages for scientific computing in python (Numpy, Pandas, Matplotlib, Scipy, Sympy, etc.) , Database access, Regular Expressions.

Text Books/ Reference Books/ Online Resources:

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

Course Code: CS5356	Knowledge Engineering Lab	Credits 0-0-2: 1

Pre-requisites: Nil

Course Outcomes:

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

CO1	Build data cubes
CO2	Implement data preprocessing techniques on data.
CO3	Implement OLAP operations and multi-dimensional modeling
CO4	Implement data mining algorithms

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2	2	3	2	1	1
CO2	2	1	3	1	2	1
CO3	2	2	3	2	2	1
CO4	2	1	3	2	2	1

1 - Slightly; 2 - Moderately; 3 – Substantially

Syllabus:

This laboratory provides hands on exposure on building of warehouse, analysing the data using OLAP tools, and implementation of mining techniques using mining tools like SPSS, Weka etc.

Text Books/ Reference Books/ Online Resources:

1. Kendal Simon et.al, Introduction to Knowledge Engineering, SPRINGER INDIA, 2009.

Course Code: CS5357	Software Testing Lab	Credits 0-0-2: 1

Pre-requisites: Nil

Course Outcomes:

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

CO1	Analyze various test tools
CO2	Generate test cases using automated testing
CO3	Perform Functional testing and Load/Stress testing
CO4	Analyze open source testing tools for databases, Web applications and networks.

Course Articulation Matrix:

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	1	-	-	-	3	-
CO2	2	2	-	1	3	-
CO3	1	-	-	2	3	-
CO4	1	2	1	2	3	-

1 - Slightly; 2 - Moderately; 3 – Substantially

Syllabus:

Case studies on Different Testing Tools – Simulate Verification and Validation Environment – Implementing the structured system Techniques – Simulate a software testing Suite which performs the functionalities of different Phase testing of SDLC – Using of Testing Tools to carry out the Functional Testing, Load/Stress Testing – Using any automated testing Tools to Automate Testing – Using of Open Source Testing Tools for databases, Web applications and Networks etc.

Text Books / Reference Books / Online Resources:

1. Rodney C.Wilson, Unix Test Tools and Benchmarks
2. Dr. KVKK Prasad, Software Testing Tools, Covering WinRunner, Silk Test, Load Runne, Jmeter, Test Director and QTP with case studies, Dreamtech Press

Course Code: CS6301	Cryptography and Network Security	Credits 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Analyze encryption algorithms.
CO2	Apply cryptographic algorithms to build secure protocols
CO3	Identify system vulnerabilities of communication protocols
CO4	Design of secure protocols to solve real world scenario

Course Articulation Matrix:

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	1	-	1	-	2	-
CO2	2	2	-	3	3	1
CO3	-	-	3	1	1	2
CO4	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;

Text Books/ Reference Books/ Online Resources:

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.

Course Code: CS6302	Internet of Things	Credits 3-0-0: 3

Pre-requisites: Nil

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

CO1	Analyze the protocol Stack for Internet of Things to address the heterogeneity in devices and networks
CO2	Develop smart IoT Applications using smart sensor devices and cloud systems
CO3	Development of smart mobile apps for societal applications
CO4	Design secure protocols for IoT systems

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2	2	2	3	-	-
CO2	3	-	3	-	3	2
CO3	3	3	-	3	-	2
CO4	2	3	2	2	2	-

1 - Slightly; 2 - Moderately; 3 – Substantially

Syllabus:

Internet of Things-Concepts, Characteristics, Challenges, M2M and Smart IoT Technology, Applications of IoT, IoT Architectures- Reference Model, Protocol Architecture, Overview of WSN-Sensor Technology, RFID, Sensor-based Solutions and RFIDs in IoT. Network Technology- Data Link Layer Protocols- 802.11, 802.15.1, 802.15.4, ZigBee, BLE and 6LoWPAN and other Communication Standards, Internet Layer Protocols- IPv6 Protocol, Addressing, BLE Over IPv6, ZigBee Over IPv6 and 802.11 Over IPv6, Internet Layer-Routing Protocols- RPL, AODV, Transport Protocols- UDP in IoT, Application Layer Communication Protocols- MQTT, CoAP, AMQT and other Protocols, Review of Edge Computing and Fog Computing, Cloud Computing in IoT, Smart Applications-Smart City, Health-Care.

Text Books / Reference Books / Online Resources:

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

Course Code: CS6303	IoT And Network Security Lab	Credits 0-0-3: 1.5

Pre-requisites: CS5304-Computer Communications and Network

Course Outcomes:

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

CO1	Analyse the protocol Stack for Internet of Things to address the heterogeneity in devices and networks
CO2	Usage of real IoT protocols for communication
CO3	Design IoT network that transfer the data to the end user using application protocols
CO4	Design an IoT network to work with a Cloud Computing infrastructure

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5
CO1	2	-	2	3	-
CO2	2	2	2	3	3
CO3	2	2		3	3
CO4	2	2	2	2	2

1 - Slightly; 2 - Moderately; 3 – Substantially

Syllabus:

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 different two UDP clients and the 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 the verify the correctness of the received packets. (Hint: Use 6LowPAN Module in NS3 and Validation using Wireshark, Use NetAnim for Visualization).
3. Design an IoT Network that consists of three RPL Border Router. Each RPL Border Router has two different edge network and each RPL Border Router is used to connect a regular IP network with a RPL 6LoWPAN edge 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 and Use Web-Sense Motes and Sky Motes).
4. Design an MQTT-SN that exchanges the messages between the node and the broker of the network. Create a simulation with a RPL Border Router device to connect a regular IP network with a 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 the data, transmit the data through internet and stores into IoT Cloud. To perform this activity create a network topology of 10 Nodes and establish the 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 Router. Each RPL Border Router has two different edge network and each RPL Border Router is used to connect a regular IP network with a RPL 6LoWPAN edge network. Store the generated data from the edges devices into IoT Cloud. To perform this activity create a network topology of 10 Nodes and establish the connection between them, Display the data and visualise 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

Text Books / Reference Books / Online Resources:

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

Course Code: CS6304	Machine Learning	Credits 3-0-0: 3

Pre-Requisites: Nil

Course Outcomes:

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

CO1	Understand instance-based learning algorithms
CO2	Design neural network to solve classification and function approximation problems
CO3	Build optimal classifiers using genetic algorithms
CO4	Build optimal classifiers using Bayesian learning

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2	1	1	2	1	1
CO2	2	3	2	2	1	1
CO3	2	3	2	2	1	1
CO4	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 - General-to-specific ordering of hypotheses, Find-S, List then eliminate algorithm, Candidate elimination algorithm, Inductive bias - decision tree learning - Decision tree learning algorithm-Inductive bias- Issues in Decision tree learning; - artificial neural networks – Perceptrons, Gradient descent and the Delta rule, Adaline, Multilayer networks, Derivation of backpropagation rule-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, EM algorithm; - 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 Neighbor Learning, Locally Weighted Regression, Kernel Methods - Dual Representations, Constructing Kernels, Radial basis function networks, Gaussian Processes for Regression and Classification; Case-based learning - genetic algorithms – an illustrative example, Hypothesis space search, Genetic Programming, Models of Evolution and Learning; Sparse Kernel Machines- Maximum Margin Classifiers, Multi class SVMs, reinforcement learning - The Learning Task, Q Learning, Nondeterministic rewards and actions, Temporal difference learning, Generalizing from examples, relationship to Dynamic Programming

Text Books / Reference Books / Online Resources:

1. Tom Mitchell, *Machine Learning*, McGraw Hill International; Edition.
2. Christopher M.Bishop, *Pattern Recognition and Machine Learning*, Springer Publishing

Course Code: CS6305	Machine Learning Lab	Credits 0-0-2: 1

Pre-requisites: Nil

Course Outcomes:

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

CO1	Implement Adaline and use for playing 2 player game
CO2	Build a neural network to solve classification problems
CO3	Build optimal classifiers using genetic algorithms
CO4	Develop Perceptron for linearly separable problems

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	2	2	2	1	1
CO2	2	3	2	2	2	1
CO3	2	3	2	2	2	1
CO4	2	3	2	2	1	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 Gabil's method of using genetic algorithm to obtain the classifier for the 2 input EXOR gate.
7. Design and implement genetic algorithm to learn conjunctive classification rules for the **Play-golf** problem.
8. Implement the Candidate-Elimination Algorithm for any given dataset.

Text Books / Reference Books / Online Resources:

1. Tom M.Mitchell, Machine
2. Learning, McGrawHill,1997.
3. SS Shwartz and SBDavid, Undersatanding Machine Learning from theory to algorithms, Cambridge University Press, 2018
4. Nikhil Buduma, Fundamentals of Deep Learning Designing Next-Generation Machine Intelligence Algorithms, Oreilly, 2017

Course Code: CS5311	Artificial Intelligence	Credits 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Solve searching problems using A*, Mini-Max algorithms.
CO2	Create logical agents to do inference using first order logic.
CO3	Understand Bayesian Networks to do probabilistic reasoning.
CO4	Perform Statistical learning using EM algorithm.

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	3	3	3	3	2
CO2	3	3	2	3	3	2
CO3	3	3	2	2	2	2
CO4	3	3	2	2	2	1
CO5	3	3	3	3	3	2

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.

Text Books/ Reference Books/ Online Resources:

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.

Course Code: CS5312	Information Systems Management	Credits 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Determine key terminologies and concepts including IT, marketing, management, economics, accounting, finance in the major areas of business.
CO2	Design, develop and implement Information Technology solutions for business problems.
CO3	Analysis of computing systems and telecommunication networks for business information systems.
CO4	Understand ethical issues that occur in business, evaluate alternative courses of actions and evaluate the implications of those actions.
CO5	Plan projects, work in team settings and deliver project outcomes in time.

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	1	1	2	2	1	1
CO2	2	3	2	1	2	1
CO3	2	1	-	1	1	2
CO4	2	1	1	2	2	1
CO5	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.

Text Books/ Reference Books/ Online Resources:

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

Course Code: CS5313	Information Security	Credits 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Evaluate the risks and vulnerabilities in protocols/Standards.
CO2	Apply Number Theory and Algebra required for designing cryptographic algorithms
CO3	Design symmetric key and asymmetric key encryption techniques.
CO4	Design authentication, message integrity and authenticated encryption protocols.
CO5	Design and analyze security of systems

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2	1	-	1	1	1
CO2	3	1	-	1	1	1
CO3	2	2	-	2	2	1
CO4	2	3	-	2	2	1
CO5	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

Text Books/ Reference Books/ Online Resources:

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

Course Code: CS5314	Modeling and Simulation	Credits 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Understand the concept of probability theory
CO2	Analyze different distributions
CO3	Understand the concept of Poisson process, Marko chains and Stochastic process
CO4	Identify different building blocks of simulation and analyze simulation results

Course Articulation Matrix:

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	-	-	-	-	-
CO2	3	1	1	-	-	-
CO3	3	2	1	-	-	-
CO4	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.

Text Books/ Reference Books/ Online Resources:

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.

Course Code: CS5315	Advanced Web Technologies	Credits 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Define the fundamental ideas and standards underlying Web Technology;
CO2	Differentiate the major frameworks allowing to develop web applications
CO3	Develop business processes using the Workflows
CO4	Develop and Deploy web applications using appropriate open source technologies
CO5	Discuss concepts at the frontier of industrial practice and emerging standards

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2	2	3	2	2	2
CO2	2	3	2	3	3	2
CO3	2	2	2	2	2	2
CO4	2	2	2	3	3	2
CO5	1	2	2	2	2	2

1 - Slightly; 2 - Moderately; 3 – Substantially

Syllabus:

HTML5- XML (Extensible Markup Language)- DTDs, Customized Markup Languages, XML parsers- XSL- Web2.0- JQuery, CSS3- Responsive Web Design- Java Server Faces Technologies- RESTFUL Web services- JSON- AJAX- NodeJS- Secure Web Application Development-3D Web Design- Search Engines- Web3.0- Next Generation Web Technologies.

Text Books/ Reference Books/ Online Resources:

1. Eric Van der Vlist, Danny Ayers, Eric Bruchez, Joe Fawcett and Alessandro Vernet, *Professional Web 2.0 Programming*, Wrox Professional, 2006.
2. Deitel, Deitel & Nieto, *Internet and Worldwide Web - How to Program*, 5th Edition, PHI, 2011.
3. Kogent Sol inc, *HTML5 Black Book*, Dreamtech Press. 2010.
4. Andreas Anyuru, *Professional WebGL Programming: Developing 3D Graphics for the Web*, Wrox Professional, 2012.

Course Code: CS5361	Cloud Computing	Credits 3-0-0: 3

Pre-requisites: CS5304-Computer Communications and Network, CS5302-Operating System Concepts

Course Outcomes:

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

CO1	Determine Cloud Computing Architectures and Services for various societalapplications
CO2	Analyze Cloud infrastructure including Google Cloud and Amazon Cloud.
CO3	Develop private and hybrid cloud for organizations to execute customizedapplications
CO4	Analyze authentication, confidentiality and privacy issues in Cloud computing environment.
CO5	Determine financial and technological implications for selecting cloud computing platforms

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	3	2	3	3	1
CO2	2	2	-	3	2	-
CO3	3	3	-	3	3	1
CO4	1	1	1	1	3	1
CO5	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

Text Books/ Reference Books/ Online Resources:

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

Course Code: CS5362	Distributed Operating Systems	Credits 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Analyze the hardware and software issues in modern distributed system.
CO2	Understand distributed architecture, naming, synchronization, consistency and replication, fault tolerance, security and distributed file system.
CO3	Analyze shared memory techniques.
CO4	Understand various file access methods.
CO5	Implementation of synchronization and deadlock.

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	3	2	3	2	1
CO2	3	3	2	3	3	1
CO3	2	2	1	2	2	1
CO4	2	2	1	2	2	1
CO5	3	3	2	2	3	1

1 - Slightly; 2 - Moderately; 3 – Substantially

Syllabus:

Introduction to distributed systems – Communication models indistributed systems – Synchronization – Processor allocation – Scheduling algorithms – Distributed file systems – Distributed sharedmemory – Case Studies of Distributed systems.

Text Books/ Reference Books/ Online Resources:

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.

Course Code: CS5363	Ubiquitous Computing	Credits 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Understand the implications of the term ubiquitous computing
CO2	Understand the current state of the art of the technologies supporting ubiquitous computing
CO3	Appreciate the challenges that exist in achieving ideal ubiquitous computing scenario
CO4	Gain an appreciation of various application domains and utility of ubiquitous computing

Course Articulation Matrix:

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	1	-	-	1	-	-
CO2	2	1	-	2	-	1
CO3	2	1	-	1	1	2
CO4	3	2	-	1	1	2

1 - Slightly; 2 - Moderately; 3 – Substantially

Syllabus:

Elements of Ubiquitous computing – Architecture – New devices - Overview of Mobile Technologies - Anatomy of a Mobile Device - Application Design Elements - Mobile Web - Development Environments - Objective-C - The Model-View-Controller Model - The Delegate Pattern - The HTML5, iPhone, Android, & Blackberry SDKs-iOS - Windows Mobile - Cellular networks - Wireless (802.11) - TCP/IP in the mobile setting- The iPhone Human Interface Guidelines- Common User Interface Guidelines - Distributed Computing - Security Issues - Upcoming Technologies - Convergence of Media and Communication Devices

Text Books/ Reference Books/ Online Resources:

1. Stefan Poslad, *Ubiquitous Computing: Smart Devices, Environment and Interactions*, John Wiley & Wiley, 2009.
2. Frank Adelstein, Sandeep K S Gupta, Golden G Richard III and Loren Schwiebert, *Fundamentals of Mobile and PervasiveComputing*, MGH, 2005.
3. T. Mikkonen, *Programming Mobile Devices: An Introductionfor Practitioners*, Wiley, 2007.
4. S. Hashimi, S. Komatineni, D. MacLean, *Pro Android 2*, Apress, 2010.

Course Code: CS5364	Image Processing	Credits 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Classify Image representations
CO2	Apply Image transformation methods
CO3	Implement image processing algorithms
CO4	Design object detection and recognition algorithms
CO5	Recover the information, knowledge about the objects in the scene

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2	3	3	2	1	1
CO2	2	1	2	2	1	1
CO3	1	2	2	2	1	1
CO4	2	3	2	2	1	1
CO5	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.

Text Books/ Reference Books/ Online Resources:

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

Course Code: CS5365	Parallel Computing	Credits 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Design and analyze parallel algorithms for real world problems and implement themon available parallel computer systems.
CO2	Optimize the performance of a parallel program to suit a particular hardware andsoftware environment.
CO3	Write Programs using accelerator technologies of GPGPUs with CUDA, OpenCL.
CO4	Design algorithms suited for Multicore processor systems using OpenCL, OpenMP,and Threading techniques.
CO5	Analyze the communication overhead of interconnection networks and modify the algorithms to meet the requirements.

Course Articulation Matrix:

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2	3	2	2	2	1
CO2	2	2	1	2	3	1
CO3	1	2	2	2	2	2
CO4	2	3	-	2	2	1
CO5	2	2	1	2	3	1

1 - Slightly; 2 - Moderately; 3 – Substantially

Syllabus:

Parallel Programming platforms – Parallel Algorithm – Basic Communication Operations – Analytical Modeling of Parallel Programs – Programming using MPI – Matrix, graph and sortingalgorithms.

Text Books/ Reference Books/ Online Resources:

1. Ananth Grama, Anshul Gupta and Vipin Kumar, *Introductionto Parallel Computing, 2/e*, Pearson Edition 2009.
2. Michael J Quinn, *Parallel Computing Theory and Practice*, 2/e,TMH, 2002.

Course Code: CS5366	E-Commerce Technologies and Management	Credits 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Understand E-commerce frameworks and advantages
CO2	Acquire in-depth knowledge of e-commerce business models and managing an enterprise
CO3	To demonstrate clear, concise, thoughtful and good understanding of electronic payment system and its types
CO4	To demonstrate a good understanding of e-marketing and its types
CO5	Comprehend the future trends in e-commerce

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	1	3	2	3	3	2
CO2	2	3	2	3	3	2
CO3	2	3	3	2	3	2
CO4	2	3	3	2	3	1
CO5	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

Text Books/ Reference Books/ Online Resources:

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

Course Code: CS5367	Network Programming	Credits 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Understand the advanced knowledge of networking
CO2	Understand advanced knowledge of programming for network communications
CO3	Analyze various solutions to perform inter-process communication
CO4	Understand the implication of security issues in networking

Course Articulation Matrix:

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	2	1	3	2	1
CO2	2	2	1	3	2	1
CO3	2	1	1	2	2	1
CO4	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

Text Books/ Reference Books/ Online Resources:

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

Course Code: CS5368	GPU Programming	Credits 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Analyze for the performance of GPU memory hierarchy
CO2	Develop parallel programs using OpenCL library
CO3	Generate parallel programs for matrix, graph and sorting problems using Cuda library
CO4	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 / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2	2	1	2	2	1
CO2	2	3	1	2	2	1
CO3	2	3	1	2	2	1
CO4	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

Text Books/ Reference Books/ Online Resources:

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

Course Code: CS5369	Software Architecture	Credits 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Understand the architecture, creating it and moving from one to any, different structural patterns
CO2	Analyze the architecture and build the system from the components
CO3	Design creational and structural patterns
CO4	Learn about behavioral patterns
CO5	Do a case study in utilizing architectural structures

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	1	-	-	3	3	2
CO2	1	2	-	2	3	2
CO3	1	3	-	1	3	2
CO4	1	-	-	2	3	2
CO5	2	-	-	3	3	2

1 - Slightly; 2 - Moderately; 3 – Substantially

Syllabus:

Introduction and architectural drivers**,** Introduction –Architecture Standard Definitions – Architectural structures – Influence of softwarearchitecture 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 structures.

Text Books/ Reference Books/ Online Resources:

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.

Course Code: CS5370	Randomized Algorithms	Credits 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Design and analyze efficient randomized algorithms
CO2	Apply tail inequalities to bound error-probability
CO3	Analyze randomized algorithms with respect to the probability of error and expected running time.
CO4	Analyze approximation algorithms and determine approximation factor.

Course Articulation Matrix:

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2	1	1	1	2	1
CO2	2	1	-	1	1	1
CO3	2	1	1	1	2	1
CO4	2	1	-	-	2	1

1 - Slightly; 2 - Moderately; 3 – Substantially

Syllabus:

Las Vegas and Monte Carlo Algorithms, Computational Model andComplexity Classes, Game Tree Evaluation, The Markov and Chebyshev's Inequality, The Stable Marriage Problem, The Coupon Collectors Problem, The Chernoff Bound, Routing in a Parallel Computer, The Probabilistic Method: Overview, probablistic analysis, use of indicator random variables, Randomly permuting arrays, Birthday paradox, analysis using indicator random variables, Balls and bins, Streaks, Online hiring problem, Maximum Satisfiability, Expanding Graphs, The Lovasz Local Lemma, Markov Chains, Random Walks on Graphs, Graph Connectivity, Expanders and Rapidly Mixing Random Walks, Pattern Matching, Random Treaps, Skip Lists, Hash Tables, Linear Programming, The Min-Cut Problem, Minimum Spanning Trees, The DNF Counting Problem, The Online approximations paging Problem, Adversary Models and Paging against an Oblivious Adversary, Randomized number theoretic and algebraic algorithms

Text Books/ Reference Books/ Online Resources:

1. Rajeev Motwani and Prabhakar Raghavan, *RandomizedAlgorithms*, Cambridge University Press, 1995.
2. J. Hromkovic, *Design and Analysis of Randomized Algorithms*, Springer, 2005.

Course Code: CS5371	User Interface Design	Credits 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Understand the characteristics of Graphical and Web User Interfaces
CO2	Use the principles of the development process
CO3	Apply the interaction styles
CO4	Analyze the design issues

Course Articulation Matrix:

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	1	1		1	1	1
CO2	1	1		2	2	2
CO3	2	2		2	2	2
CO4	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.

Text Books/ Reference Books/ Online Resources:

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

Course Code: CS6311	Digital Forensics	Credits 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Understand the need for digital forensics
CO2	Identify different technologies for digital forensics
CO3	Understand different investigation methodologies
CO4	Apply the digital forensics for different fields.

Course Articulation Matrix:

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2	2	1	2	3	2
CO2	1	3	2	2	2	2
CO3	1	2	1	2	2	2
CO4	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.

Text Books / Reference Books / Online Resources:

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

Course Code: CS6312	Human Computer Interaction	Credits 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Understand the usability of interactive systems.
CO2	Apply the techniques to manage the design process.
CO3	Use the appropriate interaction style for a given problem.
CO4	Design an interface for a given scenario, based on the concepts of HCI.

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	1	1		1	1	1
CO2	2	2		2	2	2
CO3	2	2		2	2	2
CO4	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.

Text Books / Reference Books / Online Resources:

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.

Course Code: CS6313	Design Patterns	Credits 3-0-0:3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Understand common design patterns in the context of incremental/iterative development
CO2	Evaluate and retractor software source code using patterns
CO3	Analyze and combine design patterns to work together in software design
CO4	Implement the design patterns in an object oriented language.

Course Articulation Matrix:

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2	3	2	2	2	1
CO2	1	2	2	2	3	3
CO3	2	3	3	2	2	2
CO4	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

Text Books / Reference Books / Online Resources:

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

Course Code: CS6314	Big Data Technologies	Credits 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Analyze big data challenges in different domains including social media, transportation, finance and medicine
CO2	Explore relational model, SQL and capabilities of emergent systems in terms of scalability and performance
CO3	Apply machine learning algorithms for data analytics
CO4	Analyze the capability of No-SQL systems

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2	1	3	2	2	1
CO2	1	1	3	2	2	1
CO3	2	2	3	2	2	1
CO4	2	2	3	2	2	1

1 - Slightly; 2 - Moderately; 3 – Substantially

Syllabus:

Overview of Big Data, Stages of analytical evolution, State of the Practice in Analytics, The Data Scientist, Big Data Analytics in Industry Verticals, Data Analytics Lifecycle, Operationalizing Basic Data Analytic Methods Using R, Advanced Analytics - Analytics for Unstructured Data - Map Reduce and Hadoop, The Hadoop Ecosystem, In-database Analytics, Data Visualization Techniques, Stream Computing Challenges, Systems architecture, Main memory data management techniques, Energy-efficient data processing, Benchmarking, Security and Privacy, Failover and reliability.

Text Books / Reference Books / Online Resources:

1. Bill Franks, *Taming The Big Data Tidal Wave*, 1st Edition, Wiley, 2012.

2. Frank J. Ohlhorst, *Big Data Analytics*,1st Edition, Wiley, 2012

Course Code: CS6315	BIOINFORMATICS	Credits 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Understand the theoretical basis behind bioinformatics.
CO2	Compute homologues, analyse sequences, construct and interpret evolutionary trees.
CO3	Analyse protein sequences, identify proteins, and retrieve protein structures from databases.
CO4	Understand homology modeling and computational drug design.

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	1	2	1	1	1	1
CO2	1	1	1	1	2	1
CO3	1	2	2	1	2	1
CO4	1	1	2	1	1	1

1 - Slightly; 2 - Moderately; 3 – Substantially

Syllabus:

Introduction to Bioinformatics: Define DataBase, Types of Databases, Biological Databases, Pitfalls of Biological Databases, Information Retrieval from Biological Databases, Pairwise 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, Exhaustive Algorithms, Heuristic Algorithms, Gene Prediction, Categories of gene prediction programs, Gene prediction in prokaryotes and Eukaryotes, Phylogenetics Basics Molecular phylogenetics and molecular basics Gene phylogeny versus species phylogeny, Forms of tree representation, Why finding a true tree is difficult, Phylogenetic tree construction methods and programs Protein structure basics: Amino acid, peptide formation, structure, Determination of protein 3-D structure, Protein structure database, Genome mapping, assembly and comparison, Genome mapping, Genome sequencing, Genome sequence assembly, Genome Annotation, Comparative genomics, Functional Genomics, Sequence-based approaches, Microarray-based approaches, Comparisons of SAGE and DNA microarray.

Text Books / Reference Books / Online Resources:

1. JinXiong, *Essential Bioinformatics*, 1st Edition, Cambridge University Press, 2011.
2. Arthur M Lesk, *Introduction to Bioinformatics*, 2nd Edition, Oxford University Press, 2007.

Course Code: CS6316	ERP and Supply Chain Management	Credits 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Understand the important role of ERP and Supply chain management
CO2	Implement ERP and analyze its product life cycle
CO3	Apply supply chain theories, practices and concepts utilizing case problems and problem based learning situations
CO4	Analyze supply chain modelling strategies for policy management

Course Articulation Matrix:

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	-	2	1	1	2	3
CO2	-	2	1	-	-	1
CO3	-	2	2	-	-	1
CO4	-	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.

Text Books / Reference Books / Online Resources:

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.

Course Code: CS6317	Web Analytics	Credits 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Understand, analyze and build dynamic and interactive websites
CO2	Understand current and evolving Web languages for integrating media and user interaction in both frontend and back end elements of a Web site
CO3	Analysis and reporting of web data using web analytics
CO4	Applying different testing and debugging techniques and analyzing the website effectiveness.

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	1	2	-	2	2	1
CO2	1	2	-	2	1	2
CO3	1	2	-	1	1	1
CO4	1	3	-	2	2	2

1 - Slightly; 2 - Moderately; 3 – Substantially

Syllabus:

Web Analytics Fundamentals - Capturing Data - Monitoring Web Usability - Clickstream Analysis - Website Traffic Analysis- Search and Keyword Analysis- Audience Identification and Segmentation Analysis- Web Analytics Tools.

Text Books / Reference Books / Online Resources:

1. Avinash Koushik, *Web Analytics:An Hour A Day*, Wiley Publications, 2007.
2. Avinash Koushik, *Web Analytics 2.0: The art of online accountability & Science of Customer Centricity*, Wiley Publications, 2010.
3. Justin Curtroni, *Google Analytics*, O'reilly Media, 1st Edition, 2010.
4. Pedro Sostre and Jennifer Le Claire, *Web Analytics for Dummies*, Wiley, 1stEdition 2007.

Course Code: CS6318	Algorithmic Graph Theory	Credits 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Analyze time and space complexities of algorithms using asymptotic analysis.
CO2	Formulate and solve graph problems.
CO3	Identify algorithm design methodology to solve problems.
CO4	Design efficient polynomial-time algorithms for restricted classes of intractable problems.

Course Articulation Matrix:

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2	2	-	1	2	1
CO2	3	2	1	1	1	1
CO3	2	3	1	1	2	1
CO4	2	3	-	1	2	1

1 - Slightly; 2 - Moderately; 3 – Substantially

Syllabus:

Introduction, Degree sequences, Havel–Hakimi algorithm, Erdös- Gallai theorem, Cheriton-Tarjan algorithm for minimum spanning tree, Connectivity, Blocks, Algorithms for finding the blocks of a graph, Matching, Hungarian algorithm for maximum matching, Perfect matchings and 1-factorizations, Network flows, Ford-Fulkerson algorithm, Matchings and flows, Hamiltonian cycle, Euler cycle, Chinese postman problem, Strong components, Tournaments, 2- Satisfiability, Perfect graphs, Greedy graph coloring algorithm, Some special classes of graphs, Algorithms for recognition of chordal graphs, Unit disk graphs, Dominating sets, Complexity of dominating and connected dominating sets, Algorithms for dominating sets, Hardness of some graph problems, Algorithms for independent sets and cliques, Random graphs, Social network models.

Text Books / Reference Books / Online Resources:

1. Martin Charles Golumbic, *Algorithmic Graph Theory and Perfect Graphs*, Academic Press,1980.
2. M.E.J.Newman, *Networks:An Introduction*, OxfordUniversityPress,2010.
3. TeresaW.Haynes, Stephen T. Hedetniemi and Peter J. Slater, *Fundamentals of Dominationin Graphs*, Marcel Dekker, Inc., 1998.
4. William Kocay and Donald L. Kreher, *Graphs, Algorithms, and Optimization*, CRC Press, 2005.

Course Code: CS6319	Program Analysis and Verification	Credits 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Apply the theory of abstract interpretation.
CO2	Examine existing techniques
CO3	Combine algorithms for program analysis
CO4	Experiment with Soot and Java software packages.

Course Articulation Matrix:

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2	-	3	1	1	-
CO2	2	-	2	2	1	-
CO3	2	-	3	2	1	-
CO4	1	-	2	3	3	-

1 - Slightly; 2 - Moderately; 3 – Substantially

Syllabus:

Introduction - Nature of Program Analysis, Data Flow Analysis, Equational Approach, and Constraint-Based, Type and Effect Systems, Effect Systems, Algorithms.

Data Flow Analysis – Intraprocedural Analysis, Available Expressions Analysis, Reaching Definitions Analysis, Very Busy Expressions Analysis, Live Variables Analysis, Structural Operational Semantics, Correctness of Live Variables Analysis, Monotone Frameworks, Equation Solving, Interprocedural Analysis, Shape Analysis.

Constraint Based Analysis - Abstract 0-CFA Analysis, Theoretical Properties, Constraint Based 0-CFA Analysis, Adding Context Information.

Abstract Interpretation – Correctness, Approximation of Fixed Points, Galois Connections, Induced Operations.

Type and Effect Systems - Control Flow Analysis, Theoretical Properties, Inference Algorithms, Effects, Behaviours.

Algorithms - Worklist Algorithms, Iterating in Reverse Postorder

Text Books / Reference Books / Online Resources:

1. Flemming Nielson, Hanne R. Nielson and Chris Hankin, *Principles of Program Analysis*, Springer, 2005.
2. Edmund M.Clarke, Jr., Orna Grumberg, Daniel Kroening, Doron Peled and Helmut Veith, *Model Checking*, MITPress, Second Edition,2018
3. Aaron R.Bradley and Zohar Manna, *The Calculus of Computation*, Springer, 2007
4. Daniel Kroening and ofer Strichman, *Decision Procedures: An Algorithmic Point of View*, Springer, 2008.

Course Code: CS6320	Foundations of Data Science	Credits 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Apply statistical methods to data for inferences.
CO2	Analyze data using Classification, Graphical and computational methods.
CO3	Understand Data Wrangling approaches.
CO4	Perform descriptive analytics over massive data.

Course Articulation Matrix:

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2	2	3	1	1	1
CO2	2	3	3	2	2	1
CO3	2	2	2	2	2	1
CO4	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.

Text Books / Reference Books / Online Resources:

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)

Course Code: CS6321	Foundations of Blockchain	Credits 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Familiarize the functional/operational aspects of crypto currency Ecosystem
CO2	Understand emerging abstract models for Blockchain Technology
CO3	Explore platforms such as Ethereum, Zcashto, build applications on blockchain
CO4	Design and implement new ways of using blockchain for applications other than cryptocurrency

Course Articulation Matrix:

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	1	1	1	1	1	1
CO2	1	1	1	1	1	1
CO3	1	2	1	2	3	1
CO4	2	2	1	2	2	1

1 - Slightly; 2 - Moderately; 3 – Substantially

Syllabus:

The consensus problem – Asynchronous Byzantine Agreement – AAP protocol and its analysis – Nakamoto Consensus on permission-less, nameless, peer-to-peer network – Abstract models for blockchain – Garay model – RLA Model – Proof of work as random oracle – formal treatment of consistency, liveness and fairness – protocol for Stake based chains – Hybrid models.

Cryptographic basics for cryptocurrency - Overview of hashing, signature schemes, encryption schemes and elliptic curve cryptography Bitcoin – Wallet – Blocks – Merkle Tree – Hardness of mining – transaction verifiability- anonymity –forks – double spending – mathematical analysis of properties of Bitcoin. Ethereum – Ethereum Virtual Machine – wallets for Ethereum – Solidarity – Smart Contracts – Attacks on smart contracts. Other applications of Blockchain. Zero-knowledge proofs and protocols in Blockchain – Succcinctonn-interactive argument for Knowledge (SNARK) – pairing in elliptic curves- Zcash.

Text Books / Reference Books / Online Resources:

1. Arvind Narayanan, J. Bonneau, E Felten, A Miller, and S Goldfeder, *Bitcoin and Cryptocurrency Technologies*: A comprehensive Introduction, Princeton University Press, 2016
2. Relevant Research papers

Course Code: CS6322	Game Theory	Credits 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Analyze games based on complete and incomplete information about the players
CO2	Analyze games where players cooperate
CO3	Compute Nash equilibrium
CO4	Apply game theory to model network traffic

Course Articulation Matrix:

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	1	1	-	1	1	1
CO2	1	2	-	2	2	2
CO3	1	2	-	1	2	2
CO4	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

CoalitionalGameTheory: 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,

Text Books / Reference Books / Online Resources:

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

Course Code: CS6323	Quantum Computing	Credits 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Demonstrate fundamental concepts of quantum computing
CO2	Explain the basic architecture of Quantum computing using qubit
CO3	Analyze the applications of Quantum Computing Algorithms
CO4	Develop applications using Quantum Programming

Course Articulation Matrix:

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2	1			2	2
CO2	2	1		1	2	2
CO3	1	1	1	3	3	1
CO4	2	2	1	3	3	1

1 - Slightly; 2 - Moderately; 3 – Substantially

Syllabus:

Mathematics: Complex Numbers; Complex Vector Spaces Leap from Classical to Quantum: Classical Deterministic Systems; Probabilistic Systems, Quantum Systems, Assembling Systems

Basic Quantum Theory: Quantum States, Observables; Measuring; Dynamics; Assembling Quantum Systems Architecture: Bits and Qubits; Classical Gates; Reversible Gates; Quantum Gates

Algorithms: Deutsch's Algorithm; Deutsch–Jozsa Algorithm; Simon's Periodicity Algorithm; Grover's Search Algorithm; Shor's Factoring Algorithm

Programming Languages: Programming in a Quantum World; Quantum Assembly Programming; Toward Higher-Level Quantum Programming; Quantum Computation before Quantum Computers

Text Books / Reference Books / Online Resources:

1. Noson S. Yanofsky, Mirco A. Mannucci, *Quantum Computing for Computer Scientists*, Cambridge University Press, 1st Edition, 2008
2. Eleanor G. Rieffel, Wolfgang H. Polak, *Quantum Computing: A Gentle Introduction*, MIT Press, 2011

Course Code: CS6324	Social Media Analytics	Credits 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Understand the importance of social media and networks
CO2	Enhance analytical skills for analyzing social media and networking data
CO3	Develop skills to leverage extended enterprise data
CO4	Create real-life case studies using social media data

Course Articulation Matrix:

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	-	1	-	1	-	1
CO2	1	3	2	1	-	1
CO3	1	2	2	1	1	1
CO4	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

Text Books / Reference Books / Online Resources:

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.

Course Code: CS6325	Malware Detection and Mitigation	Credits 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Possess the skills necessary to carry out independent analysis of modern malware samples using both static and dynamic analysis techniques.
CO2	Have an intimate understanding of executable formats, Windows internals and API, and analysis techniques.
CO3	Extract investigative leads from host and network based indicators associated with a malicious program
CO4	Apply techniques and concepts to unpack, extract, decrypt, or by pass new antianalys is techniques in future malware samples.

Course Articulation Matrix:

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	1	2	2	2	3	2
CO2	2	3	1	2	2	2
CO3	1	2	1	2	2	2
CO4	1	2	1	2	2	2

1 - Slightly; 2 - Moderately; 3 – Substantially

Syllabus:

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.

Text Books / Reference Books / Online Resources:

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.

Course Code: CS6326	Deep Learning	Credits 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Identify Convolutional Neural Networks models to solve Supervised Learning Problems
CO2	Design Auto encoders to solve Unsupervised Learning problems
CO3	Apply Long Shot Term Memory (LSTM) Networks for time series analysis classification problems.
CO4	Apply Classical Supervised Tasks for Image Denoising, Segmentation and Object detection problems.

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2	2	3	2	2	1
CO2	2	2	3	2	2	1
CO3	2	2	3	2	2	1
CO4	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.

Text Books / Reference Books / Online Resources:

1. Ian Good Fellow, Yoshua Benjio, Aaron, Deep Learning Courville, The MIT Press.

2. Christopher Bishop, Pattern Recognition and Machine Learning, Springer, 2006.

Course Code: CS6327	Mobile Computing	Credits 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Identify mobile computing societal applications and communication constraints in wireless environment
CO2	Analyze mobile- IPv4 and IPv6 architectures with agents and proxies.
CO3	Design MAC protocols for wireless networks.
CO4	Evaluate the performance of TCP protocols in Wireless Networks with mobile nodes.

Course Articulation Matrix:

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	1	1	1	1	1	1
CO2	2	2	-	2	2	1
CO3	1	2	-	2	3	1
CO4	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.

Text Books / Reference Books / Online Resources:

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.

Course Code: CS6328	Soft Computing Techniques	Credits 3-0-0: 3

Pre-requisites: Nil

Course Outcomes:

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

CO1	Understand the concepts of population based optimization techniques
CO2	Examine the importance of exploration and exploitation in heuristic optimization techniques to attain near-global optimal solution
CO3	Evaluate the importance of parameters in heuristic optimization techniques
CO4	Apply for the solution of multi-objective optimization

Course Articulation Matrix:

PO / CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	2	-	-	-	-
CO2	3	2	-	1	-	-
CO3	3	1	-	1	-	-
CO4	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.

Text Books / Reference Books / Online Resources:

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.