DAYANANDA SAGAR UNIVERSITY

SCHOOL OF ENGINEERING

SCHEME & SYLLABUS
FOR
BACHELOR OF TECHNOLOGY (B.Tech)

COMPUTER SCIENCE & ENGINEERING

(up to 6th Semester)

(CSE)

(With effect from 2021-22)

SCHEME – B.TECH — 2021-22 ONWARDS

III SEM - COMPUTER SCIENCE & ENGINEERING

SL	PROGRAM CODE	COURSE CODE	COURSE TITLE	CR / AU	SCHEME OF TEACHING					PREREQUISITE	
					L	T	P	S/P	C	SEM	COURSE CODE
1	103	21CS2301	DISCRETE MATHEMATICAL STRUCTURES	CR	3	–	–	–	3	*	***
2	103	21CS2302	DATA STRUCTURES	CR	3	–	–	–	3	*	***
3	103	21CS2303	DIGITAL ELECTRONICS & LOGIC DESIGN	CR	3	–	–	-	3	*	***
4	103	21CS2304	FULL STACK DEVELOPMENT	CR	2	–	2	-	3	*	***
5	103	21CS2305	COMPUTATIONAL THINKING WITH PYTHON	CR	2	–	2	-	3	*	***
6	103	21CS2306	INTRODUCTION TO NETWORKS & CYBERSECURITY	CR	3	–	–	–	3	*	***
7	103	21CS2307	DATA STRUCTURES LAB	CR	–	–	2	–	1	*	***
		21CS2308	DIGITAL ELECTRONICS & LOGIC DESIGN LAB				2		1		
9	103	21CS2309	MANAGEMENT AND ENTREPRENEURSHIP	CR	2	–	–	–	2	*	***
10	103	21CS2310	LIBERAL STUDIES – I	CR	1	-	–	–	1	*	***
					19	-	08	-	23		

CR – CREDIT, AU – AUDIT, L – LECTURE, T – TUTORIAL, P – PRACTICAL, S/P – SEMINAR/PROJECT, C – NO. OF CREDITS

SCHEME – B.TECH — 2021-22 ONWARDS

IV SEM - COMPUTER SCIENCE & ENGINEERING

SL	PROGRAM CODE	COURSE CODE	COURSE TITLE	CR / AU	SCHEME OF TEACHING					PREREQUISITE	
					L	T	P	S/P	C	SEM	COURSE CODE
1	103	21CS2401	PROBABILITY AND STATISTICS	CR	3	-	-	-	3	*	***
2	103	21CS2402	DESIGN AND ANALYSIS OF ALGORITHMS	CR	3	-	-	-	3	*	***
3	103	21CS2403	PRINCIPLES OF MICROPROCESSORS AND COMPUTER ORGANIZATION	CR	4	-	-	-	4	*	***
4	103	21CS2404	FINITE AUTOMATA & FORMAL LANGUAGES	CR	3	-	2	-	4	*	***
5	103	21CS2405	SOFTWARE ENGINEERING AND PROJECT MANAGEMENT	CR	3	-	-	-	3	*	***
6	103	21CS2406	LINUX PROGRAMMING & SCRIPTTING	CR	2	-	2	-	3	*	***
7	103	21CS2407	DESIGN AND ANALYSIS OF ALGORITHMS LAB	CR	-	-	2	-	1	*	***
8	103	21CS2408	MICROPROCESSORS LABORATORY	CR	-	-	2	-	1	*	***
9	103	21CS2409	SPECIAL TOPICS – I	CR	-	-	-	4	2	*	***
10	103	21CS2410	LIBERAL STUDIES – II	CR	1	-	-	-	1	*	***
					19	-	08	04	25		

CR – CREDIT, AU – AUDIT, L – LECTURE, T – TUTORIAL, P – PRACTICAL, S/P – SEMINAR/PROJECT, C – NO. OF CREDITS

SCHEME - B.TECH - 2021-22 ONWARDS

V SEM - COMPUTER SCIENCE & ENGINEERING

SL	PROGRAM CODE	COURSE CODE	COURSE TITLE	CR/ AU	SCHEME OF TEACHING					PREREQUISITE	
					L	T	P	S/P	C	SEM	COURSE CODE
1	103	21CS3501	DATABASE MANAGEMENT SYSTEMS	CR	3	-	-		3	*	***
2	103	21CS3502	OBJECT ORIENTED DESIGN AND PROGRAMMING	CR	3	-	-	-	3	*	***
3	103	21CS3503	OPERATING SYSTEMS	CR	3	1	-	-	4	*	***
4	103	21CS3504	MACHINE LEARNING	CR	3	-	-	2	4	*	***
5	103	21CS35XX	PROFESSIONAL ELECTIVE-1	CR	3	-	-	-	3	*	AS INDICATED IN ELECTIVE LIST
6	103	21OEXXXX	OPEN ELECTIVE-1	CR	3	-	-	-	3	*	***
7	103	21CS3505	DATABASE MANAGEMENT SYSTEMS LABORATORY	CR	-	-	2	-	1	*	***
8	103	21CS3506	OPERATING SYSTEMS LABORATORY	CR	-	-	2	-	1	*	***
9	103	21CS3507	SPECIAL TOPICS -II	CR		-	-	4	2	*	***
					18	**1**	**4**	**6**	**24**		

CR - CREDIT, AU - AUDIT, L - LECTURE, T - TUTORIAL, P - PRACTICAL, S/P - SEMINAR/PROJECT, C - NO. OF CREDITS

SCHEME - B.TECH – 2021-22 ONWARDS
V SEM-PROFESSIONAL ELECTIVE - I

SL	COURSE CODE	COURSE TITLE	SCHEME OF TEACHING					PREREQUISITE	
			L	T	P	S/P	C	SEM	COURSE CODE
1	21CS3508	GRAPH THEORY	03	-	-	-	03	-	**
2	21CS3509	MICROCONTROLLERS AND EMBEDDED SYSTEMS	03	-	-	-	03	-	**
3	21CS3510	INTERNET OF THINGS	03	-	-	-	03	-	**
4	21CS3511	AGILE SOFTWARE ENGINEERING	03	-	-	-	03	-	**
5	21CS3512	DATA WAREHOUSE AND DATA MINING	03	-	-	-	03	-	**
6	21CS3513	MOBILE COMPUTING AND APPS DEVELOPMENT	03	-	-	-	03	-	**
7	21CS3514	MOOC	-	-	-	-	03	-	**

SCHEME – B.TECH – 2021–22 ONWARDS

VI SEM – COMPUTER SCIENCE & ENGINEERING

SL	PROGRAM CODE	COURSE CODE	COURSE TITLE	CR/ AU	SCHEME OF TEACHING					PREREQUISITE	
					L	T	P	S/P	C	SEM	COURSE CODE
1	103	21CS3601	COMPILER DESIGN AND SYSTEM SOFTWARE	CR	3	1	-	-	4	*	***
2	103	21CS3602	COMPUTER NETWORKS	CR	3	-	-	-	3	*	***
3	103	21CS3603	CLOUD COMPUTING	CR	3	-	-	-	3	*	***
4	103	21CS36XX	PROFESSIONAL ELECTIVE-2	CR	3	-	-	-	3	*	AS INDICATED IN ELECTIVE LIST
5	103	21CS36XX	PROFESSIONAL ELECTIVE-3	CR	3	-	-	-	3	*	
6	103	21OEXXXX	OPEN ELECTIVE-2	CR	3	-	-	-	3	*	***
7	103	21CS3604	COMPILER DESIGN AND SYSTEM SOFTWARE LAB	CR	-	-	2	-	1	*	***
8	103	21CS3605	COMPUTER NETWORKS LAB	CR	-	-	2	-	1	*	***
					18	**02**	**04**	**-**	**21**		

CR – CREDIT, AU – AUDIT, L – LECTURE, T – TUTORIAL, P – PRACTICAL, S/P – SEMINAR/PROJECT, C – NO. OF CREDITS

VI SEM-PROFESSIONAL ELECTIVE – II & III

SL	COURSE CODE	COURSE TITLE	SCHEME OF TEACHING					PREREQUISITE	
			L	T	P	S/P	C	SEM	COURSE CODE
1	21CS3606	BLOCK CHAIN AND DISTRIBUTED LEDGER	3	-	-	-	03	*	***
2	21CS3607	MACHINE LEARNING FOR HEALTHCARE	3	-	-	-	03	*	***
3	21CS3608	DEEP LEARNING	3	-	-	-	03	*	***
4	21CS3609	DIGITAL IMAGE PROCESSING	3	-	-	-	03	*	***
5	21CS3610	HUMAN COMPUTER INTERFACE	3	-	-	-	03	*	***
6	21CS3611	UG RESEARCH PROJECT-I/PRODUCT DEVELOPMENT FOUNDATION-I	-	-	-	06	03	*	***
7	21CS3612	GAME THEORY	3	-	-	-	03	*	***
8	21CS3613	DATA SCIENCE	3	-	-	-	03	*	***
9	21CS3614	BIG DATA ANALYTICS	3	-	-	-	03	*	***
10	21CS3615	QUANTUM COMPUTATION	3	-	-	-	03	*	***
11	21CS3616	MOOC	3	-	-	-	03	*	***

SEMESTER	**III**					
YEAR	**II**					
COURSE CODE	21CS2301					
TITLE OF THE COURSE	**DISCRETE MATHEMATICAL STRUCTURES**					
SCHEME OF INSTRUCTION	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	3	**-**	**-**	**-**	**39**	**3**

Perquisite Courses (if any)			
#	Sem/Year	Course Code	Title of the Course
***	***	***	***

COURSE OBJECTIVES:

- Solve problems using relations and generating functions.
- Understand and Construct mathematical arguments.
- Use propositional and predicate logic in knowledge representation and program verification.
- Develop recursive algorithms based on mathematical induction.
- Know essential concepts in graph theory and related algorithms.
- Apply knowledge of discrete mathematics in Elementary Number Theory and problem solving.

COURSE OUTCOMES:

CO No.	**Outcomes**	**Bloom's Taxonomy Level**
CO1	Classify functions, basic set theory relations.	L4
CO2	Demonstrate the correctness of an argument using propositional and predicate logic, laws and truth tables.	L2
CO3	Compare and differentiate graphs in different geometries related to edges.	L4
CO4	Apply mathematical induction, counting principles, recursion, elementary number theory.	L3
CO5	Apply and solve Euclidean Division Algorithm and Chinese Remainder Theorem.	L3

COURSE CONTENT:	
MODULE 1	**8Hrs**
RELATIONS AND FUNCTIONS: Relation and Types of relations, Closure Properties, Equivalence Relations, Partial Ordering Relations, n-ary relations, Functions: one-to-one, onto and invertible functions, sequences, indexed classes of sets, recursively defined functions, cardinality Counting Principles: Permutation, combination, the pigeon hole principle, inclusion-exclusion principle Self – Learning Component: Set theory definition and Properties	
MODULE 2	**8Hrs**
LOGIC: Propositions and truth tables, tautologies and contradictions, logical equivalence, algebra of propositions, logical implications, predicate logic, theory of inference for propositional logic and predicate logic. Introduction to Predicate Calculus.	
MODULE 3	**8Hrs**
NUMBER THEORY: Properties of Integers: Introduction, order and inequalities, absolute value, mathematical induction, division algorithm, divisibility, primes, greatest common divisor, Euclidean algorithm, fundamental theorem of arithmetic, congruence relation, congruence equations and Chinese Reminder Theorem (CRT).	
MODULE 4	**7Hrs**
GRAPH THEORY: Graphs and multi-graphs, sub-graphs, isomorphic and homomorphic graphs, paths, connectivity, Euler and Hamilton paths, labelled and weighted graphs, complete, regular and bipartite graphs, planar graphs.	
MODULE 5	**8Hrs**
TREES AND GRAPH COLORING: Trees: Definitions-properties - fundamental theorems of trees-rooted trees-binary trees-spanningtrees. Coloring of planar graphs, Chromatic Number- Chromatic partitioning- The four Color Problem-Five-color.	

TEXT BOOKS:

1. K. H. Rosen, Discrete Mathematics & its Applications, 7th Ed., Tata McGraw-Hill, 2007.
2. Narsingh Deo, Graph Theory with Applications to Engineering and Computer Science, Prentice Hall India (PHI).

REFERENCES:

1. M.Huth and M. Ryan, Logic in Computer Science, Cambridge University N.Press, 2004.

SEMESTER	**III**					
YEAR	**II**					
COURSE CODE	21CS2302					
TITLE OF THE COURSE	**DATA STRUCTURES**					
SCHEME OF INSTRUCTION	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	3	**-**	**-**	**-**	**39**	**3**

Perquisite Courses (if any)			
#	Sem/Year	Course Code	Title of the Course
*	**	***	***

COURSE OBJECTIVES:

- To introduce the concept of data structure and its applications
- To introduce C language concepts required for data structures
- To design data structure operations to solve problems
- To introduce applications of data structures
- To introduce non-primitive data structures
- To analyse the complexity of a data structure
- To introduce static and dynamic memory allocation using C language
- To explain linear data structures – stack, queue, linked list
- To explain non-linear data structures – trees and graphs

COURSE OUTCOMES:

CO No.	**Outcomes**	**Bloom's Taxonomy Level**
CO1	Outline basic C program design for data structures	L2
CO2	Implement stack & queue data structure and their applications	L3
CO3	Apply concepts of dynamic memory allocation to real-time Problems	L3
CO4	Implement tree data structure and its applications	L3
CO5	Implement graph data structure and its applications	L3
CO6	Outline the concepts of file structures	L2

COURSE CONTENT:	
MODULE 1	**7Hrs**
INTRODUCTION TO DATA STRUCTURES: Definition, Types, C Pointers, C Structure, Arrays, Representation of Linear Array in Memory, Array Operations (Insertion, Deletion, Search and Traversal), Single Dimensional Arrays, Two Dimensional Arrays, Function Associated with Arrays, Arrays as Parameters, Recursive Functions.	
MODULE 2	**9Hrs**
INTRODUCTION TO STACK AND QUEUE: **Stack:** Definition, Array Representation of Stack, Operations Associated with Stacks- Push & Pop, Applications of Stack: Recursion, Polish expressions, Conversion of Infix to Postfix, Infix to Prefix,Postfix Expression Evaluation, Tower of Hanoi. **Queue:** Definition, Representation of Queues, Operations of Queues, Priority Queues, Circular Queue.	
MODULE 3	**9Hrs**
DYNAMIC DATA STRUCTURE: **Linked List:** Types, Introduction to Singly Linked lists: Representation of Linked Lists in Memory,Traversing, Searching, Insertion & Deletion from Linked List. Doubly Linked List, Operations on Doubly Linked List (Insertion, Deletion, Traversal). Applications: Polynomial Representation & Basic Operations, Stack & Queue Implementation using Linked Lists.	
MODULE 4	**8 Hrs**
TREES & GRAPHS: **Trees:** Basic Terminology, Binary Trees and their Representation, Complete Binary Trees, Binary Search Trees, Operations on Binary Trees (Insertion, Deletion, Search & Traversal), Application: Expression Evaluation. **Graphs:** Terminology and Representations, Graphs & Multigraphs, Directed Graphs, Sequential Representation of Graphs, Adjacency Matrices, Graph Transversal	
MODULE 5	**6 Hrs**
FILE STRUCTURES: Physical storage media, File Organization, Linked Organization of File, Inverted File, Organization Records into Blocks, Sequential Blocks, Indexing & Hashing	

TEXT BOOKS:

1. A M Tannenbaum, Y Langsam, M J Augentien "Data Structures using C", Pearson, 2013
2. R.L. Kruse, B.P. Leary, C.L. Tondo, "Data Structure and Program Design in C" PHI

REFERENCES:

1. Horowitz Anderson-Freed, and Sahni, "Fundamentals of Data structures in C", 2nd Edition, Orient Longman, 2008
2. Data Structures and Algorithm analysis in C by Mark Allen Weiss, Published by Addison Wesley (3rd Indian Reprint 2000).
3. D E Knuth, The Art of Computer Programming, Volume 1, Addison-Wesley Publishing, 2013

SEMESTER	**III**					
YEAR	**II**					
COURSE CODE	21CS2303					
TITLE OF THE COURSE	**DIGITAL ELECTRONICS & LOGIC DESIGN**					
SCHEME OF INSTRUCTION	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	3	**-**	**-**	**-**	**39**	**3**

Perquisite Courses (if any)			
#	Sem/Year	Course Code	Title of the Course
***	***	***	***

COURSE OBJECTIVES:

- To understand various number systems and conversion from one to other number systems
- To introduce basic postulates of Boolean algebra
- To manipulate expressions into POS or SOP form.
- To introduce the methods for simplifying Boolean expressions like K-Map and Quine Mclusky
- To understand the concept of don't care conditions and how they can be used to further optimize the logical functions
- To design simple combinational circuits such as multiplexers, decoders, encoders
- To understand the differences between combinational and sequential Logic circuits
- To familiar with basic sequential logic component-SR Latch
- To understand the basics of various types of memories.
- To present the working of various Flip- Flops (T flip-flop, D flip-flop, R-S flip-flop, JK flip-flop)
- To get familiarized with State Diagram, State Table, State Assignment
- To design combinational circuits using programmable logic devices.
- To design sequential circuits such as different types of Counters, Shift Registers

COURSE OUTCOMES:

CO No.	**Outcomes**	**Bloom's Taxonomy Level**
CO1	Demonstrate the knowledge of binary number systems, Logic families, Boolean algebra and logic gates	L2
CO2	Analyze different methods used for simplification of Boolean expressions	L4
CO3	Design combinational logic circuits using combinational logic elements	L3
CO4	Design combinational circuits using Programmable Logic Devices	L3
CO5	Analyze sequential logic elements in the design of synchronous and asynchronous systems	L4
CO6	Design sequential systems composed of standard sequential modules, such as counters and registers	L3

COURSE CONTENT:	
MODULE 1	**9Hrs**
NUMBER SYSTEMS: BCD number representation, Unsigned and signed number representation, Binary arithmetic. **BOOLEAN ALGEBRA AND SIMPLIFICATION:** Laws of Boolean algebra, Theorems of Boolean algebra, Boolean/Switching functions and their implementation. **SIMPLIFICATION OF BOOLEAN EXPRESSIONS AND FUNCTIONS:** Canonical forms, Sum-of-Products Method, Truth Table to Karnaugh Map, Karnaugh Simplifications, Don't-care Conditions. Product-of-sums Method, Product-of-sums simplifications, Simplification by Quine-McClusky Method.	
MODULE 2	**8Hrs**
DESIGN OF COMBINATIONAL LOGIC CIRCUITS: Modular combinational logic elements- Multiplexers and Demultiplexers, Decoders, Magnitude comparator, BCD converter, Encoders, Priority encoders.	
MODULE3	**9Hrs**
INTRODUCTION TO SEQUENTIAL CIRCUITS: Introduction to Sequential Circuits. Combinational Vs sequential circuits, Clock, Clock Triggering, Memory elements and their excitation functions – Latches, T flip-flop, D flip-flop, R-S flip-flop. JK flip-flop and their excitation requirements, State diagram, state table and state equation	
MODULE 4	**6 Hrs**
REGISTERS Registers-Types of Registers, Serial In - Serial Out, Serial In - Parallel out, Parallel In - Serial Out, Parallel In –Parallel Out, Universal Shift Register. Applications of Shift Registers	
MODULE 5	**7 Hrs**
COUNTERS, PROGRAMMABLE LOGIC: Ring, Johnson counters, Design of synchronous and asynchronous Counters Programmable Logic Arrays, Design of Combinational Circuits using Programmable Logic Devices (PLDs):	

TEXT BOOKS :

1. M. Morris Mano and Michael D. Ciletti, "Digital Design", 6th Edition,N.Pearson Education, 2018

2. Donald.P. Leach, Albert Paul Malvino & Goutam Saha: Digital Principles and Applications, 8thEdition, Tata McGraw Hill, 2015

REFERENCES:

1. D Sudhaker Samuel: Illustrative Approach to Logic Design, Sanguine-Pearson, 2010.
2. Charles H. Roth: Fundamentals of Logic Design, Jr., 7th Edition, Cengage Learning, 2014
3. John M Yarbrough, "Digital Logic Applications and Design", Thomson Learning, 2006.

SEMESTER	**III**					
YEAR	**II**					
COURSE CODE	21CS2304					
TITLE OF THE COURSE	**FULL STACK DEVELOPMENT**					
SCHEME OF Instruction	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	2	**-**	**2**	**-**	**26+26**	**3**

Prerequisite Courses			
#	Sem/Year	Course Code	Title of the Course
***	***	***	***

COURSE OBJECTIVES:

1. Understand the major areas of web programming
2. To gain the skill into web applications and development.
3. To create website using HTML5, CSS3, JavaScript.
4. Server Side Scripting using Node.JS, Express JS and Mongo dB

COURSE OUTCOMES:

CO No.	Outcomes	Bloom's Taxonomy Level
CO1	Know the fundamentals of front end web technologies using HTML 5 and CSS3	L1
CO2	Apply Cascading Style Sheets and XHTML to the idea of a web application.	L3
CO3	Comprehend the principles of client-side programming and understand how to use JavaScript to implement them in order to create dynamic web sites.	L3
CO4	Implementing the principles of server side programming using Node.js, Mongo dB	L3
CO5	Applying the Node.js framework -Express.JS to create web applications faster and smarter	L3

COURSE CONTENT:	
MODULE 1: Markup Language (HTML5)	**4 Hrs**
Introduction to HTML and HTML5 - Formatting and Fonts -Commenting Code – Anchors – Backgrounds – Images – Hyperlinks – Lists – Tables – HTML Forms, Audio ,Video Tag.	
MODULE 2: CSS3	
CSS3: Levels of style sheets; Style specification formats; Selector forms; Property value forms; Font properties; List properties; Color; Alignment of text; Background images, Conflict Resolution, CSS Box Model .CSS3 features: Box Shadow, Opacity, Rounded corners, Attribute selector.	**4 Hrs**
MODULE 3 : JavaScript	**6 Hrs**
Overview of JavaScript; Object orientation and JavaScript; General syntactic, characteristics; Primitives, operations, and expressions; Screen output and keyboard input. Control statements; Arrays; Functions, Constructors; A brief introduction on pattern matching using regular expressions, DOM Events	
MODULE 4: Node JS	**6 Hrs**
Introduction to NodeJS, Set up Dev Environment, Node JS Modules, Node Package Manager, File System, Events, Database connectivity using Mongo DB.	
MODULE 5: Express.JS	**6 Hrs**
Introducing Express: Basics of Express, Express JS Middleware: Serving Static Pages ,Listing Directory Contents, Accepting JSON Requests and HTML Form Inputs, Handling Cookies.	

List of Laboratory/Practical Experiments activities to be conducted

HTML5

1. Design a web page depicting: -
    - How markup works, including the working of various basic HTML elements and attributes..
    - The basic structure of an HTML document.
    - The usage of table tag to format a web page
    - Use and tags to provide a layout to the page instead of a table Layout.
    - The usage of lists to bring order to web pages
    - The usage of other various HTML tags like Image, anchor, links etc.
2. Design a web page and embed various multimedia features in the page.
3. Building of HTML Forms

CSS3:

4. Change the Look of a web page with a Style Sheet

JAVASCRIPT

5. Design of dynamic and Interactive web pages using Java script
    - Depicting the usage of declaring variables, running loops, if/then statements, and writing functions/Constructors using JavaScript
    - Depicting Event handling using Java script.
    - Depicting the Pattern matching using regular expressions.

NODE.JS

6. Demonstrate how to use Node.js http module to create a web server.
7. Create a Node.js file that that depicts the usage of various File System Modules

EXPRESS.JS

8. Create an app that starts a server using Express.js.
9. Demonstrate the usage of various Express JS Middleware.

TEXT BOOKS:

1. Robert W. Sebesta , "Programming the World Wide Web", 7th Edition, Pearson Education, 2008.
2. Basarat Ali Syed," Beginning Node.js ",Apress ,2014

SEMESTER	**III**					
YEAR	**II**					
COURSE CODE	21CS2305					
TITLE OF THE COURSE	**COMPUTATIONAL THINKING WITH PYTHON**					
SCHEME OF Instruction	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	2	**-**	**2**	**-**	**26+26**	**3**

Perquisite Courses (if any)			
#	Sem/Year	Course Code	Title of the Course
***	***	***	***

COURSE OBJECTIVES:

- To understand basic concepts of computational thinking.
- To introduce python programming for problem solving.
- To introduce different debugging and unit testing tools.
- To solve real world problems using python data structures.
- Learn to handle files and exception handling in python.
- To explore Python's object-oriented features.
- To build Web services and Networked programs in python.
- To train students to design an application as part of the course mini- project using computational thinking with python.

COURSE OUTCOMES:

CO No.	**Outcomes**	**Bloom's Taxonomy Level**
CO1	Understand basic concepts of computational thinking.	L2
CO2	Outline basic python programming for problem solving.	L2
CO3	Apply computational thinking to solve real world programs using Python	L3
CO4	Build python programs using core data structures like list, dictionaries and tuples	L3
CO5	Implement object oriented concepts using python	L3
CO6	Design applications related to web services and network Programming.	L3

COURSE CONTENT:	
MODULE 1	**5Hrs**
INTRODUCTION TO COMPUTATIONAL THINKING AND PYTHON: Introduction to computational thinking: Stages of Computational thinking, Basics: Values, expressions and statements, Conditional execution,Functions, Iterations	
MODULE 2	**6Hrs**
PYTHON DATA STRUCTURES: Python Data Structures: Strings, Arrays, Lists, Tuples, Sets and Dictionaries	
MODULE 3	**5Hrs**
PYTHON OBJECTS: Classes and Objects: Creating classes, Using Objects, Accessing attributes, Classes as Types, Introduction to Multiple Instances, Inheritance.	
MODULE 4	**5Hrs**
EXCEPTION HANDLING: Try-Except, Exception syntax, examples, Types of exception with except, multiple exceptions with except, Try-Finally, Raise exceptions with arguments, Python built-in exceptions, User-defined exceptions, Assertions	
MODULE 5	**5Hrs**
PYTHON FILES & LIBRARIES Files: File types, modes, File functions, File attributes, File positions, Looping over file. Basics of NumPy and Pandas	

List of Laboratory/Practical Experiments activities to be conduct

1. Python program to evaluate Values, expressions, and statements, Conditional execution, and Functions Iterations
    a. prompt the user to enter an integer and reverse it. And print the sum of the reversed integer.
    b. Write a python program to find whether a number (num1) is a factor of 255.
    c. Write a python program to find whether a number (num1) is a factor of 255.
    d. Write a program to find the sum of the following series:
        i. 1 + 1/3 + 1/5 + 1/7 + ... up to 'N' terms.
        ii. 1 + x/1! + x3/2! + x5/3! + x7/4 + ... x2n-1/n!
2. Python program to evaluate Python Collections
    a. Write a Python Program to demonstrate the inbuilt functions of Strings, List, and sets.
    b. Write a Python program for counting a specific letter 'o' in a given string; the number of times vowel 'o' appears.
    c. Write a Python Program to find the frequency of each word in given strings/strings
    d. Store the following for 'n' countries, using a dictionary:
        i. Name of a country, country's capital, per capita income of the country.

ii. Write a program to display details of the country with the highest and second lowest per capita income.

3. Write a python program to create two classes “Python” and “Java” having data members “Version” and “name” and a member function “display()”. With the help of the object, print the appropriate messages.
4. Create a class “Employee” with __init__ method to initialize data members: Name, Designation, Ph. No., and a member function display(). Create an instance for the class and display the details of the employee
5. Write an interactive calculator! User input is assumed to be a formula that consist of a number, an operator (at least + and -), and another number, separated by white space (e.g. 1 + 1). Split user input using str.split(), and check whether the resulting list is valid:
    a. If the input does not consist of 3 elements, raise a FormulaError, which is a custom Exception.
    b. Try to convert the first and third input to a float (like so: float_value = float(str_value)). Catch any ValueError that occurs, and instead raise a FormulaError
    c. If the second input is not '+' or '-', again raise a FormulaError
    d. If the input is valid, perform the calculation and print out the result. The user is then prompted to provide new input, and so on, until the user enters quit.
6. Write a Python program to count the number of lines in a text file and read the file line by line and store it into a list as well as find the longest word in the file.
7. Write a Python program to create a list of student details: usn, name dob and email {using dictionary} and write a list to a file.
8. Generate one-hot encodings for an array in numpy.
9. Write a Pandas program to import excel data into a Pandas dataframe and find a list of employees where hire_date is between two specific month and year.

TEXT BOOKS:

1. “Python for Everybody-Exploring Data Using Python 3”, Dr. Charles R. Severance,
2. “Introduction to Computing & Problem Solving with Python”,Jeeva Jose,P.Sojan Lal, Khanna Book Publishing; First edition (2019).

REFERENCES:

1. “Computer Science Using Python: A Computational Problem- Solving Focus”, CharlesDierbach, Introduction John Wiley, 2012.
2. “Introduction to Computation and Programming Using Python”, John V Guttag,Prentice Hall of India, 2015.
3. "How to think like a Computer Scientist, Learning with Python", Allen Downey,JeffreyElkner and Chris Meyers, Green Tea Press, 2014.
4. “Learning to Program with Python”, Richard L. Halterman, 2011.

SEMESTER	**III**					
YEAR	**II**					
COURSE CODE	21CS2306					
TITLE OF THE COURSE	**INTRODUCTION TO NETWORKS AND CYBERSECURITY**					
SCHEME OF Instruction	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	3	**-**	**-**	**-**	**39**	**3**

Prerequisite Courses (if any)			
#	Sem/Year	Course Code	Title of the Course
***	***	***	***

COURSE OBJECTIVES:

- To introduce the fundamental aspects of various types of computer networks.
- To demonstrate the TCP/IP and OSI models with merits and demerits.
- Understand the basic concepts of cyber security, how it has evolved, and some key techniques
  used today.
- Have an insight view of Security, Cryptography, Malware, IDS, Secure Programming etc
- Explore the subject through prescribed book, case studies, seminars and Assignments.

COURSE OUTCOMES:

CO No.	Outcomes	**Bloom's Taxonomy** Level
CO1	Understand and explore the basics of Computer Networks and workingprinciples.	L2
CO2	Understand the concepts of Network security corresponding to variousInternet Layers.	L2
CO3	Determine appropriate mechanisms for protecting the Network.	L2

COURSE CONTENT:	
MODULE 1	**8Hrs**
Overview of the Internet: Protocol, Layering Scenario, TCP/IP Protocol Suite: The OSIModel, Internet Architecture; Comparison of the OSI and TCP/IP reference model. Top- down approach	

Cybersecurity: Basics of Cyber Security-Attacks, Vulnerabilities and Threats. Need for NetworkSecurity, Data Security and physical security.	
MODULE 2	**8 Hrs**
Application Layer- Introduction, providing services, Applications layer paradigms, Client server model, Standard client-server application-HTTP, DNS, SSH. Malware Detection System, Types of Malware, Viruses & Counter Measures, Worms, Bots. **E-mail Security:** PGP, S/MIME. Secure socketprogramming using UDP and TCP.	
MODULE 3	**8 Hrs**
Transport Level Security: Functionality and services, TCP and UDP basics, Principles of Cryptography, Web Security Considerations, Secure Sockets Layer (SSL), Transport Layer Security, Data/Message Integrity and Digital Signatures.	
MODULE 4	**7 Hrs**
Network Layer Security: Network Security and Services, IP Security Overview, IP Security Policy, Encapsulation Security Payload (ESP), Internet Key Exchange. Virtual Private Network(VPN), Wireless Networks Security.	
MODULE 5	**8 Hrs**
Data Link Layer: LLC and MAC Sublayer services, Error detection and correction Techniques. **Physical Layer:** Introduction to Guided transmission media and wireless transmission media. Transmission mode, Classification of networks. Firewall, Intrusion Detection System (IDS)	

TEXT BOOK:

1. Computer Networking- A top-down approach- James F Kurose and Keith W Ross,6th Edition,Pearson Education.
2. Computer Security- Principles and Practice, William Stalling, Laurie Brown 4th Edition, Pearson

REFERENCES:

1. Behrouz A. Forouzan, Data Communications and Networking -, Fifth Edition TMH, 2013.
2. Computer Networks - Andrew S Tanenbaum, 4th Edition, Pearson Education.
3. James Graham, Richard Howard, Ryan Olson- Cyber Security Essentials CRC Press.

SEMESTER	**III**					
YEAR	**II**					
COURSE CODE	21CS2307					
TITLE OF THE COURSE	**DATA STRUCTURES LAB**					
SCHEME OF Instruction	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	-	**-**	**2**	**-**	**26**	**1**

Perquisite Courses (if any)			
#	Sem/Year	Course Code	Title of the Course
*	**	**	***

COURSE OBJECTIVES:

- To introduce C language concepts required for data structures
- To design data structure operations to solve problems
- To introduce applications of data structures
- To implement linear data structures – stack, queue, linked list
- To implement non-linear data structures – trees and graphs

COURSE OUTCOMES:

CO No.	Outcomes	Bloom's Taxonomy Level
CO1	Design and develop the programs in C to understand the different concepts of data structures.	L3
CO2	Implement stack & queue data structure and their applications, Analyse the output based on the given input data.	L3
CO3	Implement Conversions of Polish and reverse polish expressions and Record Experimental process and results	L4
CO4	Apply and implement concepts of dynamic memory allocation	L3
CO5	Use the concepts of file structures and communicate results effectively	L3

Sl. No.	List of Laboratory/Practical Experiments activities to be conduct
1.	Write a program to add, subtract, multiply and divide two integers using user defined function with return type.
2.	Write a program to find the sum of digits of the number and print the reverse of that number using Recursive Function.
3.	Write a program to add and multiply two matrices using pointers

4.	Design, Develop and Implement a menu driven Program in C for the Searching Techniques on arrays i.e, 1. Linear search 2. Binary search. If unsorted array is given as input, your program must perform sorting (bubble sort) to use it as input for binary search algorithm.
5.	Write a C program to convert infix expression to prefix expression.
6. 7. 8. 9.	Write a C program to convert infix expression to postfix expression. Write a C program to implement stack, queue and their variations using arrays. Write a C program to evaluate postfix expressions Write a C program to solve tower of hanoi using recursion
10.	Write a C program to implement stack, queue and their variations using linked lists.
11. 12.	Write a C program to implement Binary search tree insertion, deletion and traversal. Write a C program to implement Graph insertion, and traversal.
13.	Write a C program to implement File operations a. Open a file b. Write c. Read d. close d. close

Open-Ended Experiments

1. A man in an automobile search for another man who is located at some point of a certain road. He starts at a given point and knows in advance the probability that the second man is at any given point of the road. Since the man being sought might be in either direction from the starting point, the searcher will, in general, must turn around many times before finding his target. How does he search to minimizethe expected distance travelled? When can this minimum expectation be achieved?

2. The computing resources of a cloud are pooled and allocated according to customer demand. This has led to increased use of energy on the part of the service providers due to the need to maintain the computing infrastructure. What data structure will you use for allocating resources which addresses the issue of energysaving? Why? Design the solution.

3. Mini-Project on applying suitable data structure to a given real-world problem

TEXTBOOKS:

1. A M Tannenbaum, Y Langsam, M J Augentien "Data Structures using C", Pearson, 2013
2. R.L. Kruse, B.P. Leary, C.L. Tondo, "Data Structure and Program Design in C" PHI

REFERENCE BOOKS

1. Horowitz Anderson-Freed, and Sahni, "Fundamentals of Data structures in C", 2nd Edition, OrientLongman, 2008

2. Data Structures and Algorithm analysis in C by Mark Allen Weiss, Published by Addison Wesley (3rdIndian Reprint 2000).
3. D E Knuth, The Art of Computer Programming, Volume 1, Addison-Wesley Publishing, 2013

SEMESTER	**III**					
YEAR	**II**					
COURSE CODE	21CS2308					
TITLE OF THE COURSE	**DIGITAL ELECTRONICS & LOGIC DESIGN LAB**					
SCHEME OF Instruction	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	-	**-**	**2**	**-**	**26**	**1**

Perquisite Courses (if any)			
#	Sem/Year	Course Code	Title of the Course
*	**	**	***

COURSE OBJECTIVES:

- To design digital circuit for given Boolean expressions using logic gates.
- To verify the design of arithmetic circuits using logic gates and ICs.
- To test different code-conversion circuits.
- Applications of Multiplexer and De-multiplexers for implementation of different logic circuits.
- To test comparator circuits.

COURSE OUTCOMES:

CO No.	Outcomes	Bloom's Taxonomy Level
CO1	Build a logic circuit using basic gates after simplifying the given Boolean expression using Karnaugh map method	L3
CO2	Design and implementation of comparators	L3
CO3	Build logic circuits and realize the given Boolean expression using Multiplexers.	L3
CO4	Design of Combinational circuits like Encoder and Decoder using basic gates	L3
CO5	Design of Synchronous and Asynchronous Sequential circuits like registers and counters.	L3

List of Laboratory/Practical Experiments activities to be conducted
1. Study and verification of Basic gates with Truth Tables 2. Simplification of expressions using Karnaugh Maps and realizing circuitsusing Basic Gates 3. Realize binary to gray code converter and vice versa 4. Simplify the given expression using tabular method and to realize circuits using Multiplexers. 5. Design and implementation parallel adder and subtractor 6. Design and implementation of comparators 7. Design various combinational logic circuits like encoders, decoders 8. Design and implementation of shift register 9. Design and implementation synchronous counters

10. Design and implementation ring counter and Johnson counter
11. Study of 7490 BCD counter
12. Design and implementation of asynchronous counters

TEXT BOOKS:

1. M. Morris Mano and Michael D. Ciletti, “Digital Design”, 6th Edition, Pearson Education, 2018
2. Donald.P. Leach, Albert Paul Malvino & Goutam Saha: Digital Principles and Applications, 8th Edition, Tata McGraw Hill, 2015

SEMESTER	**III**					
YEAR	**II**					
COURSE CODE	21CS2309					
TITLE OF THE COURSE	**MANAGEMENT & ENTREPRENEURSHIP**					
SCHEME OF INSTRUCTION	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	2		**-**	**-**	**26**	**2**

Perquisite Courses (if any)			
#	Sem/Year	Course Code	Title of the Course
***	***	***	***

COURSE OBJECTIVES:

- Identify and analyze the factors that contribute to the process of successfully launching an entrepreneurial venture and managing a new business.
- Learn the entrepreneurial process from idea generation to implementation.
- Acquaint with special problems of starting new ventures, finding products and services, which can support new enterprises, and raising capital.
- Discuss how to start own business and also to work in or with small business or are involved with entrepreneurship.

COURSE OUTCOMES:

CO No.	**Outcomes**	**Bloom's Taxonomy Level**
CO1	Demonstrate knowledge of the key elements of the entrepreneurial process	L2
CO2	Employ strategies to generate new ideas for startups	L2
CO3	Outline how to protect IP legally	L2
CO4	Examine different ways of generating funding	L2
CO5	Explain organizing managing people, finance and customers	L2

COURSE CONTENT:	
MODULE 1	**5Hrs**
OVERVIEW OF ENTREPRENEURSHIP: THE ENTREPRENEURIAL PERSPECTIVE: Nature and Development of Entrepreneurship. Defining Manager, Entrepreneur, Entrepreneurship and Entrepreneurship. Key Elements of Entrepreneurship. Personality Characteristics of Successful Entrepreneurs. Common Myths about Entrepreneurs. Ethics and Social Responsibility of Entrepreneurs. Types of Start-Up Firms. Process of New Venture Creation. Role of Entrepreneurship in Economic Development. Emerging Trends and Issues in Entrepreneurship. **Case Study: Successful Entrepreneurs Narayana Murthy Infosys**	

MODULE 2	6Hrs
THE ENTREPRENEURIAL AND ENTREPRENEURIAL MIND: The Entrepreneurial Process: Identify and Evaluate the Opportunity, Develop a Business Plan, Determine the Resources Required, Manage the Enterprise. Managerial Versus Entrepreneurial Decision Making: Strategic Orientation, Commitment to Opportunity, Commitment of Resources, Control of Resources, Management Structure, Entrepreneurial Venturing inside a Corporation, Causes for Interest in Entrepreneurship, Climate for Entrepreneurship, Entrepreneurial Leadership Characteristics. **Case study: How to develop effective Business Plan**	
MODULE 3	**5Hrs**
CREATIVITY AND BUSINESS IDEA: Identify and Recognizing Opportunities: Observing Trends and Solving Problems. Creativity: Concept,Components and Types of Creativity, Stages of Creative Process. Sources of New Venture Ideas.Techniques for Generating Ideas. Stages of Analyzing and Selecting the Best Ideas. Protecting the Idea:Intellectual Property Rights and its Components. Linking Creativity, Innovation and Entrepreneurship. **Case study: Application of Design Thinking in New business ideas generation in particular sector (Health care, Water Saving, Energy saving)**	
MODULE 4	**5Hrs**
PREPARING THE PROPER ETHICAL AND LEGAL FOUNDATION: Initial Ethical and Legal Issues Facing a New Firm, Establishing a Strong Ethical Culture, Choosing an attorney (Lawyer), Drafting a founder's agreement, Avoiding legal disputes, Choosing a form of business organization, Obtaining business licenses and permits, Choosing a Form of Business Ownership (Sole, Proprietorship, Partnership, Corporation & Limited Liability Company) **Case study: Startup Law A to Z IP** https://techcrunch.com/2019/02/25/startup-law-a-to-z-intellectual-property/	
MODULE 5	**5Hrs**
MANAGING EARLY GROWTH AND CHALLENGES Recruiting and Selecting Key Employees. Lenders and Investors. Funding Requirements: Sources of Personal Financing. Venture Capital. Commercial Banks. Sources of Debt Financing. Key Marketing Issues for New Ventures. Why marketing is critical for Entrepreneurs. Entrepreneurs face unique Marketing Challenges. Guerrilla Marketing. Business Growth: Nature of Business Growth, Planning for Growth, Reasons for Growth. Managing Growth: Knowing and Managing the Stages of Growth, Challenges of Growing a Firm. Strategies for Firms Growth: Internal and External Growth Strategies. Implications of Growth for the Firm and Entrepreneur. Entrepreneurial Skills and Strategies to Overcome Pressures On: Financial Resources (Financial Control, Managing Inventory and Maintaining Good Records). Human Resources, Management of Employees, Time Management. **Case study: 9 ways to get startups funded** https://www.quicksprout.com/how-to-get-your-startup-funded/	

TEXT BOOKS:

1. Barringer, Ireland, "Entrepreneurship: Successfully Learning New Ventures", Pearson,Latest Edition.
2. Hisrich, Peters, Shepherd, "Entrepreneurship", Mc Graw Hill, Sixth Edition.

SEMESTER	**IV**					
YEAR	**II**					
COURSE CODE	21CS2401					
TITLE OF THE COURSE	**PROBABILITY AND STATISTICS**					
SCHEME OF Instruction	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	3	**-**	**-**	**-**	**39**	**3**

Perquisite Courses (if any)			
#	Sem/Year	Course Code	Title of the Course
***	***	***	***

COURSE OBJECTIVES:

- Understand probability, random variable and random process concepts and their importance in Computer Engineering course.
- Calculate statistics related to Random variables and process such as mean, variance, etc.
- Evaluate standard distribution functions such as Poisson's, Normal distributions
- Apply functions of random variables such as characteristic function, moment generating function to calculate statistics.
- Understand probability, random variable and random process concepts and their importance in Computer Engineering course.

COURSE OUTCOMES:

CO No.	**Outcomes**	**Bloom's Taxonomy Level**
CO1	Compute and interpret descriptive statistics using numerical and graphical techniques.	L4
CO2	Understand the basic concepts of random variables and find an appropriate distribution for analyzing data specific to an experiment.	L2
CO3	Extend the concepts to multiple random variables and apply them to analyze practical problems.	L2
CO4	Make appropriate decisions using statistical inference that is the central to experimental research.	L4

COURSE CONTENT:	
MODULE 1: INTRODUCTION TO PROBABILITY THEORY:	**6 Hrs**
Basic Notions of Probability, Axiomatic definition, properties, Conditional Probability and Independence – Baye's Theorem.	
MODULE 2: DISCRETE PROBABILITY DISTRIBUTIONS:	**7 Hrs**

Discrete random variables and its properties - Bernoulli trials – Binomial Distribution and its properties – Poisson Distribution and its properties.	
MODULE 3	**8 Hrs**
CONTINUOUS PROBABILITY DISTRIBUTIONS Continuous random variables and its properties – Exponential Distribution and its properties - Normal Distribution and its properties. **BIVARIATE DISTRIBUTIONS:** Bivariate random variables – Joint – Marginal - Conditional distribution**.**	
MODULE 4: RANDOM PROCESS AND QUEUING THEORY	**9 Hrs**
Classification – Stationary process – Markov process – Markov chain – Poisson process Auto correlation functions – Cross correlation functions – Properties – Power spectral density Queuing Models, Methods for generating random variables and Validation of random numbers	
MODULE 5: TESTING OF HYPOTHESIS	**9 Hrs**
Testing of hypothesis – Introduction-Types of errors, critical region, procedure of testing hypothesis-Large sample tests- Z test for Single Proportion - Difference of Proportion, mean and difference of mean - Small sample tests- Student's t-test.	

TEXT BOOKS:

1. A First Course in Probability, S. Ross, Pearson International Edition, 9th Edition.
2. Fundamentals of Mathematical Statistics, S. C. Gupta and V. K. Kapoor, Sultan Chand & Sons, 11thEdition.

REFERENCES:

1. K. S. Trivedi, Probability and Statistics with Reliability, Queuing, and L.Computer Science Applications, 2nd Ed., Wiley, 2001.
2. Robert V. Hogg, J.W. McKean, and Allen T. Craig: Introduction to Mathematical Statistics,Seventh Edition, Pearson Education, Asia.
3. Rohatgi, V K. and Saleh , A. K. Md. Ehsanes, "An Introduction to Probability and Statistics", (John Wiley and Sons) , (2nd edition) (2000)
4. Higher Engineering Mathematics by B S Grewal, 42 nd Edition, Khanna Publishers.
5. Probability and Statistics for engineers and scientists, R.,E.Walpole, R.H.Myers, S.L.Mayers andK.Ye, 9th Edition, Pearson Education (2012).
6. An Introduction to Probability Theory and its Applications, W. Feller , Vol. 1, 3rd Ed., Wiley, 1968

SEMESTER	**IV**					
YEAR	**II**					
COURSE CODE	21CS2402					
TITLE OF THE COURSE	**DESIGN AND ANALYSIS OF ALGORITHMS**					
SCHEME OF INSTRUCTION	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	3	-	-	-	39	3

Perquisite Courses (if any)			
#	**Sem/Year**	**Course Code**	**Title of the Course**
*	**	**	***

COURSE OBJECTIVES:

- To introduce and implement various techniques for designing algorithms and advanced data structures
- To learn space and time complexity analysis of algorithms.
- To understand the Divide and conquer design strategy and the Greedy Technique
- To understand the concepts of Dynamic Programming Applications
- Synthesize efficient algorithms in common engineering design situations

COURSE OUTCOMES:

CO No.	Outcomes	Bloom's Taxonomy Level
CO1	Outline the overview of Data structures and Algorithms	L1
CO2	Understand the different Algorithmic Design strategies	L2
CO3	Apply the Design principles and concepts to Algorithmic design	L3
CO4	Describe the DAA paradigms and when an Algorithmic Design situation calls for it.	L6
CO5	Analyse the efficiency of Algorithms using Time and Space complexity theory	L4
CO6	Implement an existing algorithm to improve the run time efficiency	L3

COURSE CONTENT:	
MODULE 1: INTRODUCTION	**6 Hrs**
The role of Algorithms in Computing, Running time analysis -- recall of asymptotic notation, big-oh, theta, big-omega, and introduce little-oh and little-omega. Worst case and average case complexity	
MODULE 2: DIVIDE AND CONQUER	**9 Hrs**
Recursive algorithms, Divide-and-Conquer recurrences, Methods for solving recurrences: substitution method, recursion tree method and the Master method. Examples-Binary search, Quick sort, Merge sort, Strassen's Matrix Multiplication. **GREEDY METHOD** Optimal substructure property- Minimum cost spanning tree, Knapsack problem, Single Source Shortest Path Algorithm. Fractional knapsack	
MODULE 3 : DYNAMIC PROGRAMMING	**9 Hrs**

Integral knapsack (contrasted with the fractional variant: 0/1 knapsack), longest increasing subsequence, All pair shortest path in graph, Matrix chain multiplication, Travelling salesman Problem	
MODULE 4: APPLICATION OF GRAPH TRAVERSAL TECHNIQUES	**9 Hrs**
Recall representation of graphs, BFS, DFS, connected components, Strongly-connected components of DAGs, Kosaraju's algorithm 1 and 2, Applications. Graph matching, String Matching: Boyer Moore algorithm.	
MODULE 5: REASONING ABOUT ALGORITHMS	**6 Hrs**
Complexity Analysis (Polynomial vs Non-Polynomial time complexity), P, NP-hard and NP-Completeness, Reductions.	

TEXT BOOK:

1. T. H. Cormen, Leiserson, Rivest and Stein, "Introduction of Computer algorithm," , 3rd Edition, The MIT Press, 2015

REFERENCES:

1. E. Horowitz, S. Sahni, and S. Rajsekaran, " Fundamentals of Computer Algorithms," Galgotia Publication, 2015.
2. Anany Levitin, ―Introduction to the Design and Analysis of Algorithms, Third Edition, Pearson Education, 2012
3. Sara Basse, A. V. Gelder, "Computer Algorithms : Introduction Design & Analysis", 3rd Edition, Addison Wesley.
4. J.E Hopcroft, J.D Ullman, "Design and analysis of Computer algorithms", PearsonEducation, 2009.
5. Steven S. Skiena, "The Algorithm Design Manual", Second Edition, Springer, 2008

SEMESTER	**IV**					
YEAR	**II**					
COURSE CODE	21CS24**03**					
TITLE OF THE COURSE	**PRINCIPLES OF MICROPROCESSORS & COMPUTER ORGANIZATION**					
SCHEME OF INSTRUCTION	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	4	**-**	**-**	**-**	**52**	**4**

Perquisite Courses (if any)			
#	Sem/Year	Course Code	Title of the Course
*	*	**	***

COURSE OBJECTIVES:

- To introduce the architecture of 8086
- To understand the importance and function of each pin of 8086 Microprocessor
- To familiarize with the architecture of 8086 microprocessor and its operation
- To understand the various addressing modes required for assembly language
- Programming and to calculate the physical address.
- To learn the 8086 instruction set and write 8086 Assembly level programs
- To understand the importance of different peripheral devices and their interfacing to 8086
- Understand the concepts of Hardwired control and micro programmed control.
- To explain the current state of art in memory system design
- Discuss the concept of memory organization.
- Summarize the types of memory.
- Learn about various I/O devices and the I/O interface.
- Learn the different types of serial communication techniques.
- To understand DMA technique
- To provide the knowledge on Instruction Level Parallelism
- To understand the concepts of pipelining techniques.

COURSE OUTCOMES:

CO No.	**Outcomes**	**Bloom's Taxonomy** Level
CO1	Identify the basic building blocks of 8086 microprocessor and use the addressing modes for executing programs efficiently	L2
CO2	Develop 8086 assembly language programs using modern assembler tools	L3
CO3	Discuss the computer arithmetic and design algorithms for various Arithmetic operations.	L2
CO4	Design data part and control part of a processor	L3
CO5	Analyze the performance of various classes of Memories	L4
CO6	Understand pipeline & parallel processing	L2

COURSE CONTENT:	
MODULE 1: Introduction to Microprocessor & its Architecture:	**8 Hrs**
Introduction-Evolution of Microprocessor, The Microprocessor-Based Personal Computer Systems, Internal Microprocessor Architecture, Real mode memory addressing, 8086 pin diagram, Internal Architecture of 8086, Registers, Addressing Modes-Immediate addressing, Register addressing, direct addressing, indirect addressing, relative addressing, Instruction formats	
MODULE 2: Programming 8086	**12 Hrs**
Assembler directives, Data Movement Instructions, String Data Transfers, Miscellaneous Data Transfer Instructions, Arithmetic and Logic Instructions, BCD and ASCII Arithmetic, Basic Logic Instructions, Shift and Rotate, String Comparisons. Program Control Instructions: The Jump Group,Assembly language programming with 8086, macros, procedures	
MODULE 3: Processor Organization:	**10 Hrs**
Basic organization of computers, Block level description of the functional units as related to the execution of a program; Fetch, decode and execute cycle. Execution cycle in terms machine instructions. Information representation, Floating point representation (IEEE754), computer arithmetic and their implementation; **Data Part Design:** Fixed-Point Arithmetic-Addition, Subtraction, Multiplication and Division, Arithmetic Logic Units control and data path, data path components, design of ALU and data-path, **Control Part Design:** Control unit design; Hardwired and Micro programmed Control unit. Discussions about RISC versus CISC architectures.	
MODULE 4: Memory Technology, Input/Output Organization:	**12 Hrs**
Memory hierarchy, static and dynamic memory, RAM and ROM chips, Memory address map, Auxiliary Memory, Associative Memory, Cache Memory and organization. Peripheral devices, Input-Output Interface; I/O Bus and Interface Modules, Isolated versus Memory-Mapped I/O, Modes of Transfer; Programmed I/O, Interrupt-initiated I/O, Direct memory access (DMA)	
MODULE 5: Pipelining	**10 Hrs**
Basic Concepts, Arithmetic Pipeline, Instruction Pipeline; Four-Segment Instruction Pipeline, Pipeline hazards and their resolution, **Parallel Processing;** Flynn's classification, Multicore architectures, Introduction to Graphics Processing Units, Example: NVIDIA GPU Architecture	

TEXT BOOK:

1. Barry B Brey: The Intel Microprocessors, 8th Edition, Pearson Education, 2009
2. Mano, Morris M. Computer system architecture. Dorling Kindesley Pearson, 2005.

REFERENCES:

1. Douglas V Hall, "MICROPROCESSORS AND INTERFACING, PROGRAMMING ANDHARDWARE" TMH, 2006.
2. Kenneth J. Ayala, "The 8086 Microprocessor: Programming & Interfacing The PC",Delmar Publishers, 2007
3. Muhammad Ali Mazidi, Janice Gillispie Mazidi, Danny Causey, The x86 PC Assembly LanguageDesign and Interfacing, 5th Edition, Pearson, 2013.
4. V. Carl Hamacher, Safwat G. Zaky and Zvonko G. Vranesic , Computer Organization ,McGraw-Hillseries 2002
5. Hayes, J.P , Computer Architecture and Organization, McGraw-Hill, 1998
6. David Patterson and John Hennessey, Computer Organization and Design, Elsevier. 2008
7. Comer, Douglas. Essentials of computer architecture. Chapman and Hall/CRC, 2017.
8. Hord, R. Michael. Parallel supercomputing in MIMD architectures. CRC press, 2018.
9. Tanenbaum, Andrew S. Structured computer organization. Pearson Education India, 2016.
10. William Stallings-Computer Organization and Architecture, Seventh Edition, Pearson Education

SEMESTER	**IV**					
YEAR	**II**					
COURSE CODE	21CS2404					
TITLE OF THE COURSE	**FINITE AUTOMATA AND FORMAL LANGUAGES**					
SCHEME OF Instruction	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	3	**-**	**2**	**-**	**39+26**	**4**

Perquisite Courses (if any)			
#	Sem/Year	Course Code	Title of the Course
*	*	**	***

COURSE OBJECTIVES:

- To learn general theory of automata, properties of regular sets and regular expressions.
- To understand basics of formal languages.
- To know push-down automata, context- free languages, Turing machines.

COURSE OUTCOMES:

CO No.	Outcomes	Bloom's Taxonomy Level
CO1	Understand the basic concepts of formal languages of finite automata techniques such as DFA, NFA and E-NFA	L2
CO2	Design Finite Automata for different Regular Expressions and Languages Demonstrate the properties of regular grammar, regular language, regular expression & their relationship with finite automata	L3
CO3	Construct context free grammar for various languages. Interpret and design different PDA for a given language	L3
CO4	Construct context free, regular, Chomsky normal form grammars to design computer languages	L3
CO5	Design Turing machine to solve problems	L3

COURSE CONTENT:	
MODULE 1	**8Hrs**
Introduction to Finite Automata: Study and Central concepts of automata theory, An informal pictureof finite automata, deterministic and non-deterministic finite automata, applications of finite automata,finite automata with epsilon – transitions.	
MODULE 2	**8Hrs**
Regular expression and languages: Regular expressions, finite automata and regular expressions, algebraic laws of regular expressions. Properties of Regular Languages: closure properties of regular languages, Pumping Lemma, equivalence and minimization of automata .	

MODULE 3	9Hrs
Context – free Grammars and Languages: Context free grammars, Context-free languages,Parse trees, Ambiguity in grammars and languages, Pushdown Automata: Pushdown automation (PDA), the language of PDA, equivalence of PDA's and CFG's, DeterministicPushdown Automata	
MODULE 4	**8Hrs**
Properties of Context – Free Languages: Normal forms of context free grammars, pumping lemma for context free languages, closure properties of context free languages. Applications of CFG - such as spec of programming languages, parsing techniques, and Yacc	
MODULE 5	**6Hrs**
Introduction to Turing Machine- The Turing machine, programming techniques for Turing machine, extensions to the basic Turing machine, Chomsky hierarchy	

List of Laboratory/Practical Experiments activities to be conducted
1. Design a Program for creating machine that accepts three consecutive one. 2. Design a Program for creating machine that accepts the string always ending with 101. 3. Design a Program for Mode 3 Machine 4. Design a program for accepting decimal number divisible by 2. 5. Design a program for creating a machine which accepts string having equal no. of 1's and 0's. 6. Design a program for creating a machine which count number of 1's and 0's in a given string. 7. Design a Program to find 2's complement of a given binary number. 8. Design a Program which will increment the given binary number by 1. 9. Design a Program to convert NDFA to DFA. 10. Design a Program to create PDA machine that accept the well-formed parenthesis. 11. Design a PDA to accept WCWR where w is any string and WR is reverse of that string and C is a Special symbol. 12. Design a Turing machine that's accepts the following language an b n c n where n>0.

TEXT BOOKS:

1. Daniel I. A. Cohen, Introduction to Computer Theory, 2nd Edition, Wiley India Student Edition, 2008.
2. J.E. Hopcroft , R. Motwani, and J. D. Ullman, Introduction to Automata Theory,Languages and Computation, 3rd Edn. Pearson Education , New Delhi 2008

REFERENCES:

1. K.L.P. Misra and N. Chandrashekaran. Theory of Computer Science- Automata,Languages and Computation, 3rd Edn. PHI, New Delhi, 2007

2. C. Martin - Introduction to Languages and the Theory of Computation 2ndEdn,TMH, New Delhi, 2000.

SEMESTER	**IV**					
YEAR	**II**					
COURSE CODE	21CS2405					
TITLE OF THE COURSE	**SOFTWARE ENGINEERING AND PROJECT MANAGEMENT**					
SCHEME OF INSTRUCTION	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/ Projects Hours	Total Hours	Credits
	3	**-**	**-**	**-**	**39**	**3**

Perquisite Courses (if any)			
#	Sem/Year	Course Code	Title of the Course
***	***	***	***

COURSE OBJECTIVES:

- This course is introduced to give the students necessary knowledge.
- Understanding and Design aspects in Software Engineering
- To understand the Software Project Planning and Evaluation techniques

COURSE OUTCOMES:

CO No.	**Outcomes**	**Bloom's Taxonomy Level**
CO1	Understand software development life cycle models, process models, and various design engineering techniques	L2
CO2	Apply new software models, techniques and technologies to bring out innovative and novelistic solutions for the growth of the society in all aspects and evolving into their continuous professional development	L3
CO3	Analyze a problem, and identify and define the computing requirements appropriate to its solution	L4
CO4	Apply a wide variety of testing techniques in an effective and efficient manner.	L3
CO5	Understand Project Management principles while developing software.	L2

COURSE CONTENT:	
MODULE 1	**8Hrs**
Introduction to Software Engineering: FAQs about software engineering, Professional and ethical responsibility. Socio-Technical systems: Emergent system properties, Organizations, people and computer systems; Legacy systems, the evolving role of software, Changing Nature of Software, Software myths. **A Generic view of process:** Software engineering- A layered technology, a process framework, The Capability Maturity Model Integration (CMMI), Process patterns, process assessment, personal and team process models. Software Cost Estimation: Productivity; Estimation techniques	
MODULE 2	**8Hrs**
Process models: A simple safety- critical system; System dependability; Availability and reliability, the waterfall model, Incremental process models, Evolutionary process models, The Unified process. Agile Development: Agile Tech, Extreme Programming, and other Agile Process Models: Scrum Methodology	
MODULE 3	**8Hrs**
Software Requirements: Functional and non-functional requirements, User requirements, System requirements, Interface specification, the software requirements document. Requirements engineering process: Feasibility studies, Requirements elicitation and analysis, Requirements validation, Requirements management.	
MODULE 4	**8Hrs**
Testing Strategies: Verification and Validation: Planning; Software inspections; Automated static analysis; Verification and formal methods. A strategic approach to software testing, System testing, the art of Debugging; Component testing; Test case design; Test automation - Selenium, Test strategies for conventional software: Black-Box and White-Box testing, Validation tests, System testing.	
MODULE 5	**7Hrs**
Software Project Management Introduction to Software Project Management – all life cycle activities – Methodologies – Categorization of Software Projects – Setting objectives – Management Principles – Management Control – Project portfolio Management – Cost-benefit evaluation technology – Risk evaluation – Strategic program Management – Stepwise Project Planning.	

TEXT BOOKS:

1. Software Engineering, by Ian Sommerville Eighth edition, International Computer Science Series.
2. Software Engineering, A practitioner's Approach- Roger S. Pressman, 6th edition. McGraw Hill International Edition.

REFERENCES:

1. Bob Hughes, Mike Cotterell and Rajib Mall: Software Project Management – FifthEdition, Tata McGraw Hill, New Delhi, 2012.
2. SoftwareEngineering-K.K.Agarwal&YogeshSingh,NewAgeInternationalPublishers
3. Software Engineering, an Engineering approach-JamesF. Peters, Witold Percy, John Wiley.
4. Systems Analysis and Design –Shelly Cashman Rosenblatt, Thomson Publications.
5. Software Engineering principles and practice-Waman Jawadekar,The McGraw-HillCompanies

SEMESTER	**IV**					
YEAR	**II**					
COURSE CODE	21CS2406					
TITLE OF THE COURSE	**LINUX PROGRAMMING AND SCRIPTING**					
SCHEME OF Instruction	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	2	**-**	**2**	**-**	**26+26**	**3**

Perquisite Courses (if any)			
#	Sem/Year	Course Code	Title of the Course
***	***	***	***

COURSE OBJECTIVES:

- To learn the fundamentals of OS
- To Know the features of Linux OS and learn the Linux commands
- To gain knowledge about the Linux networking and Linux administration
- To Understand the fundamentals of shell scripting/programming.
- To discuss about the Inter Process Communication.
  To understand the concept of client server communication by using sockets

COURSE OUTCOMES:

CO No.	**Outcomes**	**Bloom's Taxonomy Level**
CO1	Describe the operating system fundamentals, Linux architecture and features of Linux OS	L2
CO2	Demonstrate proficiency with Linux utilities, networking and administration	L2
CO3	Develop Shell Scripts for automation of various tasks	L3
CO4	Implement Inter-Process communication between processes	L3
CO5	Design various client server applications using TCP or UDP protocols	L3

COURSE CONTENT:	
MODULE 1: INTRODUCTION TO OS	**5Hrs**
Operating System Objectives and Functions, The Evolution of Operating Systems developments Leading to Modern Operating Systems, Virtual Machines, OS Design Considerations for Multiprocessor and Multicore architectures, Modern UNIX Systems, Linux, Booting Process of Linux operating system	
MODULE 2: INTRODUCTION TO LINUX	**6Hrs**

LINUX BASICS: File System of the Linux, basic commands, Linux users and group, Permissions for file, directory and users, Filters: cut, tr, grep. Find Command with various options, Filters using Regular Expression: grep & sed **Networking Tools:** TCP/IP basics, Resolving IP addresses, ping, telnet and ftp, cron commands	
MODULE 3: SHELL PROGRAMMING	**5Hrs**
Types of Shells, Shell Meta Characters - $#, $*, $?, Shell Variables, Shell Scripts. read, operators, Integer Arithmetic and String Manipulation, Decision Making: if-else-elif-fi, case-esac. Loop Control; while, for, until, break & continue, Functions, I/O Redirection and Piping	
MODULE 4: INTERPROCESS COMMUNICATION	**5Hrs**
Introduction to IPC, Pipes, FIFOs, Introduction to three types of IPC-message queues, semaphores and shared memory, Message Queues- Kernel support for messages, Unix system V APIs for messages, client/server example.	
MODULE 5: SOCKETS	**5Hrs**
Introduction to Sockets, Socket Addresses, Socket system calls for connection oriented protocol and connectionless protocol, example client/server programs.	

List of Laboratory/Practical Experiments activities to be conduct
1. Study and Practice on various commands like man, passwd, tty, clear, date, cal, cp, mv, ln, rm, unlink, mkdir, rmdir, du, df, mount, umount, find, unmask, ulimit, ps, who 2. Study and Practice on various commands like cat, tail, head , sort, nl, uniq, grep, egrep, fgrep, cut, paste, join, tee, pg, comm, cmp, diff, tr, tar, cpio 3. Implement in C the following Unix commands using System calls A.cat B.ls C.mv 4. Creating shell programs for automating tasks, file handling, trapping signals 5. Implement the following IPC forms a) FIFO b) PIPE 6. Write a C program (sender.c) to create a message queue with read and write permissions to write 3 messages to it with different priority numbers. 7. Write a C program (receiver.c) that receives the messages (from the above message queue as specified and displays them 8. Write client and server programs (using c) for interaction between server and client processes using TCP Elementary functions. 9. Write client and server programs (using c) for interaction between server and client processes using UDP Elementary functions

TEXT BOOKS:

1. Unix System Programming using C++, T.Chan, PHI.
2. Unix Concepts and Applications, 4th Edition, Sumitabha Das, TMH.

REFERENCES:

1. Unix Network Programming ,W.R.Stevens,PHI.
2. Andrew S. Tanenbaum, “Modern Operating Systems”, Second Edition, Pearson Education, 2004.

SEMESTER	**IV**					
YEAR	**II**					
COURSE CODE	21CS2407					
TITLE OF THE COURSE	**DESIGN AND ANALYSIS OF ALGORITHMS LABORATORY**					
SCHEME OF INSTRUCTION	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	-	-	2	-	26	1

Perquisite Courses (if any)			
#	Sem/Year	Course Code	Title of the Course
*	****	****	****

COURSE OBJECTIVES:

- To learn mathematical background for analysis of algorithm
- To understand the concept of designing an algorithm.
- To analyze the algorithms using space and time complexity.
- To learn dynamic programming and greedy method.
- To acquire knowledge of various applied algorithms.

COURSE OUTCOMES:

CO No.	**Outcomes**	**Bloom's Taxonomy Level**
CO1	Design and develop the Algorithms to understand the different concepts.	L3
CO2	Apply the Design principles and concepts to Algorithmic design	L3
CO3	Describe the DAA paradigms and when an Algorithmic Design situation calls for it.	L6
CO4	Analyse worst-case and best – case running times of algorithms using asymptotic analysis.	L4
CO5	Implement an existing algorithm to improve the run time efficiency	L3

List of Laboratory/Practical Experiments activities to be conducted
1. Design a C program to solve the Tower of Hanoi. Compute the time complexity.
2. Apply divide and conquer method and Design a C program to search an element in a given array and Compute the time complexity. Binary search - recursive method
3. Apply Divide and Conquer method Design a C program to sort an array using Merge sort algorithm and compute its time complexity

4. Apply Divide and Conquer method Design a C program to sort an array using Quick sort algorithm and compute its time complexity.
5. Apply Greedy method and Design a C program to find the minimum cost spanning tree using Prim's and Kruskal's Algorithm and compute its complexity
6. Apply Dynamic Programming Technique and Design a C program to find the all pairs shortest path using Dijkstra's Algorithm and computes its complexity
7. Design a C program to find the optimal solution of 0-1 knapsack problem using dynamic programming and Compute the time complexity
8. Design a C program to solve the Travelling Salesman Problem using dynamic programming and compute its time complexity.
9. Design a C program to find the longest common subsequence using dynamic programming and compute its time complexity
10. Mini project proposal should be submitted and Implementation should be done based on the problem stated in the proposal

TEXT BOOK:

1. Levitin A, "Introduction to the Design And Analysis of Algorithms", Pearson Education, 2008.
2. T. H. Cormen, Leiserson, Rivest and Stein, "Introduction of Computer algorithm," , 3rd Edition, The MIT Press, 2015

REFERENCES:

1. E. Horowitz, S. Sahni, and S. Rajsekaran, "Fundamentals of Computer Algorithms," Galgotia Publication, 2015.
2. Goodrich M.T., R Tomassia, "Algorithm Design foundations Analysis and Internet Examples", John Wiley and Sons, 2006.
3. Sara Basse, A. V. Gelder, "Computer Algorithms : Introduction Design & Analysis", 3rd Edition, Addison Wesley.

SEMESTER	**IV**					
YEAR	**II**					
COURSE CODE	21CS2408					
TITLE OF THE COURSE	**MICROPROCESSORS LAB**					
SCHEME OF Instruction	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	-	**-**	**2**	**-**	**26**	**1**

Perquisite Courses (if any)			
#	Sem/Year	Course Code	Title of the Course
*	**	**	****

COURSE OBJECTIVES:

- To develop and execute variety of assembly language programs of Intel 8086 including arithmetic and logical, sorting, searching, and string manipulation operations
- To develop and execute the assembly language programs for interfacing Intel 8086 with peripheral devices.

COURSE OUTCOMES:

CO No.	Outcomes	Bloom's Taxonomy Level
CO1	Implement 8086 assembly language programs for microprocessor application using 8086 training boards	L3
CO2	Implement 8086 assembly language programs for microprocessor application using assembler and debuggers	L3
CO3	Design interfacing of various peripherals with 8086 microprocessor for simple applications	L3
CO4	Use Macros and Procedures in 8086 Programs	L3
CO5	Use assembly language and debugging tools when writing programs for a microprocessor	L3
CO6	Communicate effectively on the work done in the laboratory using formal report	L3

List of Laboratory/Practical Experiments activities to be conducted

Part-A: Software Programs Using Microprocessor Trainer Kit

i) Programs involving : arithmetic operations, sorting

ii) Programs on : code conversion (BCD TO HEX, Binary to ASCII, Binary to Gray)

iii) Programs involving - Bit manipulation like checking:

1. Whether given data is positive or negative
2. Whether given data is odd or even
3. Logical 1"s and 0"s in a given data

Part- B: Software Programs Using MASM/TASM software i) Programs on : searching and sorting ii) Programs on : palindrome, string comparison iii) Programs on : current time display, Decimal up-counter display
Part-C: Hardware Programs to interface microprocessor with various peripherals Using Microprocessor Trainer Kit i) DC Motor Interface ii) Stepper Motor Interface iii) Matrix Keypad Interface iv) 7 Segment Display Interface

TEXT BOOKS:

1. Microprocessor and Interfacing - Douglas V Hall, SSSP Rao, 3rd edition TMH, 2012.
2. The Intel Microprocessor, Architecture, Programming and Interfacing - Barry B. Brey, 6e,Pearson Education / PHI, 2003.

SEMESTER	V					
YEAR	III					
COURSE CODE	21CS3501					
TITLE OF THE COURSE	DATABASE MANAGEMENT SYSTEMS					
SCHEME OF Instruction	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/ProjectsHours	Total Hours	Credits
	3	-	-	-	39	3

Perquisite Courses (if any)			
#	Sem/Year	Course Code	Title of the Course
***	***	***	***

COURSE OBJECTIVES :

- To learn data models, conceptualize and depict a database system using ER diagram
- To understand the internal storage structures in a physical DB design
- To know the fundamental concepts of transaction processing techniques

COURSE OUTCOMES

CO No.	Outcomes	Bloom's Taxonomy Level
CO1	Demonstrate the basic elements of a relational database management system	L2
CO2	Identify the data models for relevant problems	L2
CO3	Apply normalization for the development of application software's	L3
CO4	Use Structured Query Language (SQL) for database manipulation.	L3
CO5	Understand transactions and their properties (ACID)	L2
CO6	Design and develop a large database with optimal query processing	L6

COURSE CONTENT:	
MODULE 1	8Hrs
Introduction: Purpose of Database System-–Views of data–data models, database management system, three-schema architecture of DBMS, components of DBMS. E/R Model - Conceptual data modeling - motivation, entities, entity types, attributes relationships, relationship types, E/R diagram notation, examples.	
MODULE 2	8Hrs
Relational Model: Relational Data Model - Concept of relations, schema-instance distinction, keys, referential integrity and foreign keys, relational algebra operators, SQL -Introduction, data definition in SQL, table, key and foreign key definitions, update behaviors. Querying in SQL, notion of aggregation,	

aggregation functions group by and having clauses.	
MODULE 3	**8Hrs**
Database Design: Dependencies and Normal forms, dependency theory –functional dependencies, Armstrong's axioms for FD's, closure of a set of FD's, minimal covers, definitions of 1NF, 2NF, 3NF and BCNF, decompositions and desirable properties of them, algorithms for 3NF and BCNF normalization, 4NF, and 5NF	
MODULE 4	**8 Hrs**
Transactions: Transaction processing and Error recovery - concepts of transaction processing, ACID properties, concurrency control, locking based protocols for CC, error recovery and logging, undo, redo, undo-redo logging and recovery methods.	
MODULE 5	**7Hrs**
Embedded SQL: triggers, procedures and database connectivity. Introduction to NoSQL	

TEXT BOOKS :

1. Silberschatz, Henry F. Korth, and S. Sudharshan, “Database System Concepts”, 5thEd, Tata McGraw Hill, 2006.
2. J. Date, A. Kannan and S. Swamynathan, “An Introduction to Database Systems”, 8thed, Pearson Education, 2006.

REFERENCES :

1. Ramez Elmasri and Shamkant B. Navathe, “Fundamentals of Database Systems”, Fourth Edition, Pearson/Addision Wesley, 2007
2. Raghu Ramakrishnan, “Database Management Systems”, Third Edition, McGraw Hill, 2003
3. S. K. Singh, “Database Systems Concepts, Design and Applications”, First T. Edition, Pearson Education, 2006

SEMESTER	**V**					
YEAR	**III**					
COURSE CODE	21CS3502					
TITLE OF THE COURSE	**OBJECT ORIENTED DESIGN AND PROGRAMMING**					
SCHEME OF INSTRUCTION	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	3	**-**	**-**	**-**	**39**	**3**

Perquisite Courses (if any)			
#	Sem/Year	Course Code	Title of the Course
***	***	***	***

COURSE OBJECTIVES:

- Understand the basic concepts of object-oriented design techniques.
- Understand the fundamentals of object-oriented programming with Java.
- Draw UML diagrams for the software system.
- Impart basics of multi-threading and database connectivity.
- Develop GUI using event handling techniques in Java.

COURSE OUTCOMES:

CO No.	Outcomes	Bloom's Taxonomy Level
CO1	Apply the concepts of object-oriented programming in software design process.	L3
CO2	Develop Java programs using Java libraries and construct to solve real-time problems.	L3
CO3	Understand, develop and apply various object-oriented features using Java to solve computational problems	L2
CO4	Implement exception handling and JDBC connectivity in Java.	L3
CO5	Build an event-oriented GUI (graphical user interface).	L6

COURSE CONTENT:	
MODULE 1	**08 Hrs**
An Overview of Object-Oriented Systems Development: Introduction; Two Orthogonal Views of the Software; Object-Oriented Systems Development Methodology; Why an Object-Oriented? Overview of the Unified Approach. **Object Basics:** Introduction; An Object-Oriented Philosophy; Objects; Objects are Grouped in Classes; **Attributes:** Object State and Properties; Object behaviour and Methods; Object Respond to Messages; Encapsulation and Information Hiding; **Class Hierarchy:** Inheritance; Multiple Inheritance; **Polymorphism;** Object Relationships and Associations: Consumer-Producer Association; Aggregation and Object Containment; **Case Study -** A Payroll Program; Object-Oriented Systems Development Life Cycle: Introduction; Software Development Process; Building High-Quality Software; Object-Oriented Systems Development: A Use Case Driven Approach; Reusability.	

MODULE 2	08 Hrs
Unified Modelling Language: Introduction; Static and Dynamic models; Why Modeling? Introduction to the UML; UML Diagrams; UML Class Diagram; Use-Case Diagram. **Introduction to Java:** Java's Magic: The Bytecode; JVM; **Object-Oriented Programming;** Simple Java programs; Two Control Statements; Lexical Issues; Data Types; Variables, Arrays and String constructors; Operators; Control Statements; Input/Output: I/O Basic; Reading console input Writing Console output.	
MODULE 3	**10 Hrs**
Introducing Classes: Class Fundamentals; objects; methods; constructors; this Keyword; Garbage Collection; finalize() method; Parameter Passing; Overloading; Access Control Keywords. Inheritance basics; method overriding; abstract classes; Packages and interfaces. Exception handling fundamentals; multiple catch; nested try statements**. Multi-Threaded Programming :**Multi-Threaded Programming: Java Thread Model; The main Thread; Creating a thread and multiple threads; Extending threads; Implementing Runnable; Synchronization; Inter Thread Communication; producer consumer problem	
MODULE 4	**08 Hrs**
Event and GUI Programming: Introducing Swing; The Origins of Swing; Swing Is Built on the AWT; Two Key Swing Features; The MVC Connection; Components and Containers; The Swing Packages; A Simple Swing Application; Event Handling; JLabel; JTextField; JButton	
MODULE 5	**05 Hrs**
Database Access: The Concept of JDBC; JDBC Driver Types; JDBC Packages; A Brief Overview of the JDBC process; Database Connection; Associating the JDBC/ODBC Bridge with the Database; Statement Objects; ResultSet.	

TEXT BOOK:

1. Bahrami A.; Object Oriented Systems Development using the Unified Modeling Language; McGraw Hill; 1999.
2. Schildt; Herbert. Java The Complete Reference; 8th Edition. US: McGraw-Hill Osborne Media; 2011.
3. Jim Keogh; J2EE: The Complete Reference; McGraw Hill Education in 2002.

REFERENCES:

1. Barclay K., J. Savage, Object Oriented Design with UML and Java, Elsevier, 2004.

2. Y. Daniel Liang, Introduction to Java Programming, 7th edition, Pearson, 2013.

SEMESTER	**V**					
YEAR	**III**					
COURSE CODE	**21CS3503**					
TITLE OF THE COURSE	**OPERATING SYSTEMS**					
SCHEME OF INSTRUCTION	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	3	**1**	**-**	**-**	**39**	**4**

Perquisite Courses (if any)			
#	**Sem/Year**	**Course Code**	**Title of the Course**
*	***	***	****

COURSE OBJECTIVES:

- To understand the basic concepts and functions of operating systems.
- To understand Processes and Threads
- To analyze Scheduling algorithms.
- To understand the concept of Deadlocks.
- To analyze various Memory and Virtual memory management, File system and storage techniques.
- To discuss the goals and principles of protection in a modern computer system.

COURSE OUTCOMES:

CO No.	**Outcomes**	**Bloom's Taxonomy Level**
CO 1	Interpret the different structures, functions, services of operating system and use operating system level virtualization to improve security, manageability and availability of today's complex software environment with small runtime and resource overhead & with minimal changes to the existing computing infrastructure	L2
CO 2	Infer the performance of various CPU scheduling algorithms to make the system more efficient, faster & fairer	L4
CO 3	Use the knowledge of synchronization hardware, semaphores, monitors to resolve process synchronization problems	L3
CO4	Identify the deadlocks using resource allocation graph & resolve the deadlocks using roll back & abort algorithm, bankers algorithm to ensure system is free from dead locks	L2, L3
CO 5	Compare & Contrast various memory management schemes to implement the virtual address & provide the memory protection	L4
CO 6	Examine the various file management techniques, disk scheduling methods for efficient resource utilization & Interpret the system, network, program threats & employ protection principles to safeguard the system resources	L2,L5

COURSE CONTENT:	
MODULE 1: OS Overview and System Structure	**8 Hrs**
Introduction to operating systems, System structures: What operating systems do; Computer	

System organization; Computer System architecture; Operating System structure; Operating System operations; Process management; Memory management; Storage management; Protection and Security; Distributed system; Computing environments. Operating System Services: User - Operating System interface; System calls; Types of system calls; System programs; Operating system design and implementation; Operating System structure; Virtual machines;
MODULE 2: Process Management **8 Hrs**
Process Management: Process concept; Process scheduling; Operations on processes. Multi-threaded Programming: Overview; Multithreading models; Threading issues. Process Scheduling: Basic concepts; Scheduling Criteria; Scheduling Algorithms.
MODULE 3: Process Coordination **8 Hrs**
Process Synchronization: The critical section problem; Peterson's solution; Synchronization hardware; Semaphores; Classical problems of synchronization; Monitors Deadlocks: Deadlocks; System model; Deadlock characterization; Methods for handling deadlocks; Deadlock prevention; Deadlock avoidance; Deadlock detection and recovery from deadlock.
MODULE 4: Memory Management **7 Hrs**
Memory Management Strategies: Background; Swapping; Contiguous memory allocation; Paging; Structure of page table; Segmentation. Virtual Memory Management: Background; Demand paging; Copy-on-write; Page replacement; Allocation of frames; Thrashing.
MODULE 5: File System and Secondary Storage Structure **8 Hrs**
File System, Implementation of File System**:** File system: File concept; Access methods; Directory structure; File system mounting; File sharing. Protection: Implementing File system: File system structure; File system implementation; Directory implementation; Allocation methods; Free space management. Mass storage structures; Disk structure; Disk attachment; Disk scheduling; Disk management; Swap space management. Protection and Security: Protection: Goals of protection, Principles of protection, System Security: The Security Problem, Program Threats, System and Network Threats.

TEXT BOOKS:

1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, Operating System Principles 8th edition, Wiley-India, 2010

REFERENCES:

1. Operating Systems-Internals and Design Principles, William Stallings, 6th Edition, Pearson Education, 2009.
2. Operating Systems: A Modern Perspective, Gary J. Nutt, Addison-Wesley, 1997

SEMESTER	**V**					
YEAR	**III**					
COURSE CODE	21CS3504					
TITLE OF THE COURSE	**MACHINE LEARNING**					
SCHEME OF Instruction	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/ Projects Hours	Total Hours	Credits
	3	**-**	**2**	**-**	**39+26**	**4**

Perquisite Courses (if any)			
#	Sem/Year	Course Code	Title of the Course
*	**	**	****

COURSE OBJECTIVES:

- Define machine learning and understand the basic theory underlying machine learning.
- To understand the working principle of Machine Learning Algorithms
- To apply various techniques of Machine Learning Algorithms
- Perform statistical analysis of machine learning techniques.

COURSE OUTCOMES:

CO No.	Outcomes	Taxonomy Level
CO1	Outline the basic concepts of Supervised, unsupervised and reinforcement machine learning algorithms.	L2
CO2	Employ probability density functions, the basics of the sampling theorem and estimating the maximum likelihood to develop new predictive models.	L4
CO3	Implement supervised learning algorithms for regression and classification problems using Machine learning tools.	L3
CO4	Apply Unsupervised Machine Learning algorithms and feature engineering techniques to develop machine learning models.	L3
CO5	Evaluate the performance of machine learning algorithms based on Accuracy, precision, sensitivity, specificity and F1 score.	L5

COURSE CONTENT:

MODULE 1: Introduction to Machine Learning	**7Hrs**
Well posed learning problems, Designing a Learning system. Introduction to AI, Machine learning and Deep learning with applications. Types of learning: supervised, unsupervised and reinforcement learning. Perspective and Issues in Machine Learning.	

Classical paradigm of solving learning problems, The learning problems--classes and types of learning, fundamental of statistical learning and its framework. Introduction to feature representation and extraction.	
MODULE 2: Mathematics for Machine Learning	**8Hrs**
Introduction to Statics Probability (joint probability, conditional probability, Bayes theorem, different distributions, univariate and multivariate Gaussian distribution, PDF, MLE, Motivation, estimating hypothesis accuracy, Basics of sampling theorem, General approach for deriving confidence intervals, Difference in error of two hypothesis, Comparing learning algorithms.	
MODULE 3: Supervised Learning	**9Hrs**
Introduction to Supervised Learning, Introduction to Perceptron model and its adaptive learning algorithms (gradient Decent and Stochastic Gradient Decent), Introduction to classification, Naive Bayes classification Binary and multi class Classification, decision trees and random forest, Regression (methods of function estimation) --Linear regression and Non-linear regression, logistic regression, Introduction To Kernel Based Methods of machine learning: K-Nearest neighbourhood , kernel functions, SVM, Introduction to ensemble based learning methods	
MODULE 4: Unsupervised Learning	**8 Hrs**
Introduction to Unsupervised Learning, Clustering (hard and soft clustering) Hierarchal clustering: K-means, Fuzzy C-Means (FCM) algorithm, Gaussian mixture models (GMM), Expectation Maximization algorithm, feature Engineering in Machine Learning, Dimensionality reduction, Linear Discriminant Analysis and Principle Component Analysis.	
MODULE 5: Model Selection	**7Hrs**
Machine Learning model validation - Confusion Matrix, Accuracy, Precision, F score, Cost function, Machine Learning Optimization algorithms: Gradient descent, stochastic GD. Regularization: Normalization and Standardization overfitting, underfitting, optimal fit, bias, variance, cross-validation.	

List of Laboratory/Practical Experiments activities to be conducted
1. Implementation of linear and logistic regression 2. Implementation of SVM, KNN, Naïve Bayes ML algorithms 3. Implementation of Decision trees, Random Forest classifiers 4. Implement ensemble algorithms. 5. Implementation of different clustering algorithms and PCA Implementation of different neural networks Capstone project in specific domains (Health care, Transportation, Telecom etc.)

TEXT BOOKS;

1. Thomas M. Mitchell, Machine Learning, McGraw- Hill, Inc. New York, ISBN: 0070428077 9780070428072.
2. Bengio, Yoshua, Ian J. Goodfellow, and Aaron Courville. "Deep learning." An MIT Press book in preparation. (2015).

REFERENCE BOOKS:

1. Ethem Alpaydin, Introduction to Machine Learning (Adaptive Computation and Machine Learning series), The MIT Press; second edition, 2009.
2. V. N. Vapnik " The Nature of statistical Learning"

SEMESTER	**V**					
YEAR	**III**					
COURSE CODE	21CS3505					
TITLE OF THE COURSE	**DATABASE MANAGEMENT SYSTEMS LAB**					
SCHEME OF Instruction	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	-	**-**	**2**	**-**	**26**	**1**

Perquisite Courses (if any)			
#	Sem/Year	Course Code	Title of the Course
*	**	**	****

COURSE OBJECTIVES :

- Understand the fundamental concepts of database management. These concepts include aspects of database design, database languages, and database-system implementation.
- To provide a strong formal foundation in database concepts, recent technologies and best industry practices.
- To give systematic database design approaches covering conceptual design, logical design and an overview of physical design.
- To learn the SQL and NoSQL database system.
- To learn and understand various Database Architectures and its use for application development.
- To programme PL/SQL including stored procedures, stored functions, cursors and packages

COURSE OUTCOMES :

CO No.	Outcomes	Bloom's Taxonomy Level
CO1	Install and configure database systems.	L3
CO2	Analyze database models & entity relationship models.	L3
CO3	Design and implement a database schema for a given problem-domain	L3
CO4	Understand the relational and document type database systems.	L2
CO5	Populate and query a database using SQL DML/DDL commands.	L3

List of Laboratory/Practical Experiments activities to be conducted
1. Design any database with at least 3 entities and relationships between them. Apply DCL and DDL commands. Draw suitable ER/EER diagram for the system.
2. Design and implement a database and apply at least 10 different DML queries for the following task. For a given input string display only those records which match the given pattern or a phrase in the search string. Make use of wild characters and LIKE operator for the same. Make use of Boolean and arithmetic operators wherever necessary.
3. Execute the aggregate functions like count, sum, avg etc. on the suitable database. Make use of built in functions according to the need of the database chosen. Retrieve the data from the database based on time and date functions like now (), date (), day (), time () etc. Use group by and having clauses.
4. Implement nested sub queries. Perform a test for set membership (in, not in), set comparison (<some, >=some, <all etc.) and set cardinality (unique, not unique).
5. Write and execute suitable database triggers .Consider row level and statement level triggers.

6. Write and execute PL/SQL stored procedure and function to perform a suitable task on the database. Demonstrate its use.
7. Write a PL/SQL block to implement all types of cursor.
8. Execute DDL statements which demonstrate the use of views. Try to update the base table using its corresponding view. Also consider restrictions on updatable views and perform view creation from multiple tables.
9. Mini project.

TEXT BOOKS :

1. Ramon A. Mata-Toledo, Pauline Cushman, Database management systems, TMGH, ISBN: IS978-0-07-063456-5, 5th Edition.

REFERENCES :

1. Dr. P. S. Deshpande, SQL and PL/SQL for Oracle 10g Black Book, DreamTech.
2. Ivan Bayross, SQL, PL/SQL: The Programming Language of Oracle, BPB Publication.
3. Dalton Patrik, SQL Server – Black Book, DreamTech Press.

SEMESTER	**V**					
YEAR	**III**					
COURSE CODE	21CS3506					
TITLE OF THE COURSE	**OPERATING SYSTEMS LAB**					
SCHEME OF INSTRUCTION	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	-	-	**2**	**-**	**26**	**1**

Perquisite Courses (if any)			
#	**Sem/Year**	**Course Code**	**Title of the Course**
*	**	***	****

COURSE OBJECTIVES:

- To learn creating process and Threads
- To implement various CPU Scheduling Algorithms
- To implement Process Creation and Inter Process Communication.
- To implement Deadlock Avoidance and Deadlock Detection Algorithms
- To implement Page Replacement Algorithms
- To implement File Organization and File Allocation Strategies

COURSE OUTCOMES:

CO No.	**Outcomes**	**Bloom's Taxonomy Level**
CO1	Implement system calls to expose the operating system's services to user programs.	L3
CO2	Develop multi-threading and CPU Scheduling algorithms to make the system more efficient, faster, and fairer.	L3
CO3	Implement process synchronization problem using semaphores for the coordination of the process interactions in an Operating System.	L3
CO4	Implement bankers algorithm for the purpose of deadlock avoidance to ensure system is in safe state.	L3
CO5	Develop the page replacement algorithms for effective management of virtual memory.	L3
CO6	Implement file organization and file allocation strategies for efficient disk space utilization.	L3

List of Laboratory/Practical Experiments activities to be conducted		
Exp. No	Division of Experiments	List of Experiments
1	System Calls	Write a C program to create a new process that exec a new program using system calls fork(), execlp() & wait()
2		Write a C program to display PID and PPID using system calls getpid () & getppid ()
3		Write a C program using I/O system calls open(), read() & write() to copy contents of one file to another file
4	Process Management	Write a C program to implement multithreaded program using pthreads
5		Write C program to simulate the following CPU scheduling algorithms a) FCFS b) SJF c) Priority d) Round Robin
6	Process synchronization	Write a C program to simulate producer-consumer problem using semaphores
7	Deadlock	Write a C program to simulate Bankers algorithm for the purpose of deadlock avoidance.
8		Write a C program to simulate deadlock detection.
9	Memory Management	Write a C program to simulate paging technique of memory management
10		Write a C program to simulate page replacement algorithms a) FIFO b) LRU c) LFU
11	I/O System	Write a C program to simulate the following file organization techniques a) Single level directory b) Two level directory
12		Write a C program to simulate the following file allocation strategies. a) Sequential b) Indexed

TEXT BOOKS:

1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, Operating System Principles 8th edition, Wiley-India, 2010

REFERENCES:

1. Operating Systems-Internals and Design Principles, William Stallings, 6th Edition, Pearson Education, 2009.

SEMESTER	**V**					
YEAR	**III**					
COURSE CODE	21CS3507					
TITLE OF THE COURSE	**Special Topics-II**					
SCHEME OF INSTRUCTION	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	-	-	-	**4**	**52**	**2**

Course objectives

1. To develop problem solving abilities

2. To build the necessary skill set and analytical abilities for developing computer based solutions for real life problems.

3. To train students in professional skills related to Software Industry

4. To prepare necessary knowledge base for research and development in Computer Science

Course Content

Following are some of the ways (but not limited to) of delivering the "Special Topics":

(i) Engaging Students in Small Batches (maximum 3/batch) in **Projects:** DSU Faculty will define and supervise a project which has a well defined scope. Students will work from requirements to delivering a prototype.

(ii) **Delivery from an Industry Expert:** An industry Expert can offer a project for around 20-25 students, clearly defining the scope. The project will have 4-5 sub-modules. Each student group will work on one sub-module from requirements gathering and analysis all the way to a working module. The sub-teams will integrate the modules and will together deliver a working prototype. The industry expert will engage all the teams on one afternoon face to face. One or two SOE faculty will also co-supervise the project.

(iii) **A Start-up company** might have a few project ideas to try out and they would engage a team of 20-25 students (in 4-5 batches) to work on these project ideas from concept to a prototype, with a close supervision from the start-up company technologist together with DSU faculty.

(iv) **Testing a new Product:** A Company has come up with a new product and they require a team of 30-40 students to thoroughly test all the features of the product and come up with a validation of the features of the product, a summary of features that fail to work and also a recommendation ona set of features that may have to be added to the product.

(v) A professor from an elite university from within India or abroad, offering a **short course** on a domain which is very current and state of art. The content has a built in project component.

(vi) A student undergoes a **on-line certification course** (such as coursera, Edx founded by Harward and MIT, MOOC, NPTEL, SWAYAM etc). Student obtains a certificate and an 'End of the Semester' exam will be conducted by the respective department.

(vii) An expert from a company offers a **3 or 4 day workshop** (on-campus or outside the campus) involving mostly hands-on and a project component and a group of students successfully complete the workshop, with well defined learning components and deliverables.

(viii) Students participate and successfully complete **a Hackathon** (of Minimum two days), conducted by a reputed institution/organization. The deliverables include the pre-hackathon components, work done during Hackathon and post-Hackathon work (if applicable).

(ix) **Industry Project:** Students in a small team of 4-5 work on a project defined by an industry (including DERBI and AIC) during a semester and successfully complete the project.

(x) **Summer Internship:** A group of students take up Summer Internship at DSU or outside, successfully complete the internship. If done within DSU, a project exhibition will also form a part of evaluation.

(xi) **Visit to a University Abroad**: A group of students participate in a well structured program in a University abroad and complete all the requirements of the university.

(xii) **Working under a Research professor** within DSU or from premium institutes such as IISc, IIT, IIITetc on a specific project/task.

SEMESTER	V
YEAR	III

COURSE CODE	GRAPH THEORY					
TITLE OF THE COURSE	21CS3508					
SCHEME OF INSTRUCTION	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	3	-	-	-	39	3

Perquisite Courses (if any)			
#	Sem/Year	Course Code	Title of the Course
*	**	***	****

COURSE OBJECTIVES:

- To Understand and explain the basic concepts of graph theory.
- To understand the concept of digraphs, Euler digraphs and Hamiltonian digraphs.
- To develop the under-standing of Geometric duals in Planar Graphs.
- To introduce the idea of coloring in graphs

.

COURSE OUTCOMES:

CO No.	Outcomes	Bloom's Taxonomy Level
CO1	Appreciate the definition and basics of graphs along with types and their examples	L2
CO2	Understand the definition of a tree and learn its applications to fundamental circuits.	L2
CO3	Know the applications of graph theory to network flows.	L2
CO4	Understand the notion of planarity and coloring of a graph.	L2
CO5	Relate graph theory to real-world problems.	L3

COURSE CONTENT:	
MODULE 1: Paths, Circuits and Graph Isomorphisms	8Hrs
Definition and examples of a graph, Subgraph, Walks, Paths and circuits; Connected graphs, disconnected graphs and components of a graph; Euler and Hamiltonian graphs, Graph isomorphisms, Adjacency matrix and incidence matrix of a graph, Directed graphs and their elementary properties	
MODULE 2: Trees and Fundamental Circuits	8Hrs
Definition and properties of trees, Rooted and binary trees, Cayley's theorem on a counting tree, Spanning tree, Fundamental circuits, Minimal spanning trees in a connected graph.	

MODULE 3: Cut-Sets and Cut-Vertices	8 Hrs
Cut-set of a graph and its properties, Fundamental circuits and cut-sets, Cut-vertices, Connectivity and separability, Network flows, 1- isomorphism and 2- isomorphism	
MODULE 4: Planar Graphs	7Hrs
Planar graph, Euler theorem for a planar graph, Various representations of a planar graph, Dual of a planar graph, Detection of planarity, Kuratowski's theorem.	
MODULE 5: Graph Coloring	8Hrs
Chromatic number of a graph, Chromatic partition, Chromatic polynomial, Matching and coverings, Four color problem	

TEXT BOOKS:

1. R. Balakrishnan & K. Ranganathan (2012). A Textbook of Graph Theory. Springer.
2. Narsingh Deo (2016). Graph Theory with Applications to Engineering and Computer Science. Dover Publications.

REFERENCES:

1. Reinhard Diestel (2017). Graph Theory (5th edition). Springer.
2. Edgar G. Goodaire & Michael M. Parmenter (2018). Discrete Mathematics with Graph Theory (3rd edition). Pearson.
3. Douglas West (2017). Introduction to Graph Theory (2nd edition). Pearson.

SEMESTER	V					
YEAR	III					
COURSE CODE	21CS3509					
TITLE OF THE COURSE	MICROCONTROLLERS AND EMBEDDED SYSTEMS					
SCHEME OF INSTRUCTION	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	3	-	-	-	39	3

Perquisite Courses (if any)			
#	Sem/Year	Course Code	Title of the Course
*	**	***	***

COURSE OBJECTIVES:

- Explain the architectural features and instructions of 32 bit microcontroller -ARM Cortex M3.
- Develop Programs using the various instructions of ARM Cortex M3 and C language for different applications.
- Identify and understand the unique characteristics and components of embedded systems
- Understand how can we interfacing different input and output devices/components to cortex M3 microcontroller
- Understanding of how Arduino Uno & Raspberry Pi work

COURSE OUTCOMES:

CO No.	Outcomes	Bloom's Taxonomy Level
CO1	Describe the architectural features and instructions of 32 bit microcontroller ARM Cortex M3.	L2
CO2	Apply the knowledge gained for Programming ARM Cortex M3 for different applications	L3
CO3	Understand the basic hardware components and their selection method based on the characteristics and attributes of an embedded system.	L2
CO4	Develop an embedded application with Cortex M3 architecture	L3
CO5	Design embedded systems using Arduino board and RasberryPi	L3

COURSE CONTENT:	
MODULE 1: ARM-32 bit Microcontroller	**8Hrs**
Microprocessors versus Microcontrollers, Different Microcontroller Architectures (CISC, RISC, ARISC), Microcontroller Types: PIC, AVR, ARM, Background of ARM and ARM Architecture: A Brief History, Architecture Versions, The Thumb-2 Technology and Instruction Set Architecture, Cortex-M3 Processor Applications, Overview of the Cortex-M3: What Is the ARM Cortex-M3 Processor, Architecture of ARM Cortex M3, Various	

Units in the architecture, General Purpose Registers, Special Registers, Exceptions and Interrupts	
MODULE 2: ARM Cortex M3 Instruction Sets and Programming:	**8Hrs**
Assembly basics, Instruction List, Instruction Descriptions: Moving Data, LDR and ADR Pseudo-Instructions, Processing Data, Call and Unconditional Branch, Decisions and Conditional Branches, Combined Compare and Conditional Branch, Conditional Execution Using IT Instructions, Instruction Barrier and Memory Barrier Instructions, MSR and MRS, More on the IF-THEN Instruction Block, SDIV and UDIV, REV, REVH, and REVSH, Reverse Bit, SXTB, SXTH, UXTB, and UXTH.	
MODULE 3: Cortex-M3 Programming	**8Hrs**
A Typical Development Flow, Using C, CMSIS: Background of CMSIS, Organization of CMSIS, Using CMSIS, Using Assembly: The Interface between Assembly and C, The First Step in Assembly Programming, Producing Outputs, The “Hello World” Example, Using Data Memory, Simple programming exercises	
MODULE 4: Embedded System Design Concepts	**8Hrs**
Introduction: Definition of Embedded System, Embedded Systems Vs General Computing Systems, Major Application Areas, Purpose of Embedded Systems, Characteristics and Quality Attributes of Embedded Systems, Core of the Embedded System: General Purpose and Domain Specific Processors, Embedded system architecture.	
MODULE 5: Embedded System Design using Raspberry Pi	**7 Hrs**
Introduction to RaspberryPi, About the Raspberry Pi board and programming (on Linux) Hardware Layout, Operating systems on RaspberryPi, Configuring raspberry Pi, Programming raspberry Pi with Python libraries.	

TEXT BOOKS:

1. Joseph Yiu, “The Definitive Guide to the ARM Cortex-M3”, 2nd Edition, Newnes, (Elsevier),2010.
2. Shibu K V, “Introduction to Embedded Systems”, Tata McGraw Hill Education Private Limited, 2nd Edition.

REFERENCES:

1. Muhammad Tahir, Kashif Javed, ARM Microprocessor Systems: Cortex-M Architecture, CRC Press 2017
2. Richard Blum, “Arduino Programming in 24 Hours”, Sams Teach Yourself, Pearson Education, 2017.

3. Vijay Madisetti and Arshdeep Bahga, "Internet of Things (A Hands-on-Approach)", 1st Edition, VPT, 2016
4. Srinivasa K G, Internet of Things,CENGAGE Leaning India, 2017

SEMESTER	**V**					
YEAR	**III**					
COURSE CODE	**21CS3510**					
TITLE OF THE COURSE	**INTERNET OF THINGS**					
SCHEME OF INSTRUCTION	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	3	**-**	**-**	**-**	**39**	**3**

Perquisite Courses (if any)			
#	Sem/Year	Course Code	Title of the Course
***	***	***	***

COURSE OBJECTIVES:

- To learn the building blocks of the Internet of Things (IoT) and their characteristics.
- To introduce the students to the programming aspects of the Internet of Things with a view toward rapid prototyping of IoT applications.
- To learn communication protocol for IoT.
- To learn Reference architectures for different levels of IoT applications.
- To learn IoT data analytics and Tools for IoT.

COURSE OUTCOMES:

CO No.	Outcomes	Bloom's Taxonomy

		Level
CO1	Use the characteristics of loT, designs of loT, addressing Management of IoT to develop an loT device.	L3
CO2	Interpret the building blocks of the loT to identify the application areas of loT and to secure IoT.	L3
CO3	Use loT protocols and Internet Connectivity Principles in practical domains of society like Smart Agriculture, Smart Cities, IoMT.	L3
CO4	Employ suitable hardware and development tool for loT based applications: Smart Cities, loMT, Smart Agriculture.	L4
CO5	Develop and program an loT device to work with Data Analytics and Cloud Computing infrastructure.	L4

COURSE CONTENT:	
MODULE 1: INTRODUCTION TO IOT	**8 Hrs**
Introduction: Concepts behind the Internet of Things, Definition, Characteristics of IoT, IoT Conceptual framework, Physical design of IoT, Logical design of IoT, Application of IoT, IoT and M2M, IoT System Management with NETCONF-YANG.	
MODULE 2: IOT ARCHITECTURE AND SECURITY	**8 Hrs**
M2M high-level ETSI architecture, IETF architecture for IoT, IoT reference model, IoT 3 Tier, and 5 tier architecture IoT Security: IoT and cyber-physical systems, IoT security (vulnerabilities, attacks, and countermeasures), Security engineering for IoT development, IoT security lifecycle	
MODULE 3 : IOT PROTOCOLS	**7 Hrs**
IoT Access Technologies: Physical and MAC layers, Web Communication Protocols for connected devices, SOAP, REST, HTTP Restful, and Web Sockets. Internet Connectivity Principles: Internet Connectivity, Internet-based communication, Network Layer: IP versions, IP addressing in IoT, Zigbee,6LoWPAN, Routing over Low Power and Lossy Networks.	
MODULE 4 : HARDWARE AND DEVELOPMENT TOOLS FOR IOT	**8 Hrs**
Sensors, actuators, radio frequency identification (RFID) technology, wireless sensor networks, and participatory sensing technology. Embedded Platforms for IoT: Embedded computing basics, Overview of IoT supported Hardware platforms such as Arduino, Raspberry Pi, NodeMCU, Programming with Arduino, NodeMCU, and Raspberry Pi	
MODULE 5 : CASE STUDY AND REAL-WORLD APPLICATION	**8 Hrs**
Case Studies: Smart Agriculture, IoMT, Smart Cities (Smart Parking, Smart Lighting, Smart Road, Health and Lifestyle), Data Analytics for IoT, Cloud Storage Models & Communication APIs, Cloud for IoT, Amazon Web Services for IoT	

TEXT BOOK:

1. Arshdeep Bahga and Vijay Madisetti, Internet of Things - A Hands-On Approach
2. Rajkamal," Internet of Things", Tata McGraw Hill publication

REFERENCES:

1. Hakima Chaouchi "The Internet of Things: Connecting Objects", Wiley publication.
2. IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of

Things, by David Hanes, Gonzalo Salgueiro , Patrick Grossetete , Robert Barton, Jerome Henry by CISCO
3. Donald Norris "The Internet of Things: Do-It-Yourself at Home Projects for Arduino, Raspberry Pi and Beagle Bone Black", McGraw Hill publication

SEMESTER	V					
YEAR	III					
COURSE CODE	21CS3511					
TITLE OF THE COURSE	AGILE SOFTWARE ENGINEERING					
SCHEME OF Instruction	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/ProjectsHours	Total Hours	Credits
	3	-	-	-	39	3

Perquisite Courses (if any)			
#	Sem/Year	Course Code	Title of the Course
***	***	***	***

COURSE OBJECTIVES:

- Agile methodology, Scrums, Sprints.
- Agile testing, test automation, DevOps.

COURSE OUTCOMES:

CO No.	Outcomes	Bloom's Taxonomy Level
CO1	Compare and contrast the differences between Agile and other project management methodologies	L4
CO2	Interpret and apply various principles, phases and activities of the Scrum methodology	L3
CO3	Define the benefits of using an Agile approach to managing projects	L2
CO4	Understand Agile Testing principles for real life situations and learn the basics of SAFe for scaled agile	L2
CO5	Identify and use various tools for Agile development and DevOps principles for CI/CD	L3

COURSE CONTENT:

MODULE 1	8Hrs
INTRODUCTION TO AGILE : Introduction to Software engineering, SDLC, Software process models- waterfall, V model, Iterative model, Spiral model; Introduction to Agile: Agile versus traditional method comparisons and process tailoring; Introduction to Agile, Various Agile methodologies -Scrum, XP, Lean, and Kanban, Agile Manifesto.	
MODULE 2	8Hrs
SCRUM AND SPRINT: Scrum: Scrum process, roles - Product Owner, Scrum Master, Team, Release manager, Project Manager, product manager, architect, events, and artifacts; Product Inception: Product vision, stakeholders, initial backlog creation; Agile Requirements – User personas, story mapping, user stories, 3Cs, INVEST, acceptance criteria, sprints, requirements, product backlog and backlog grooming; Test First Development; Pair Programming and Code reviews;	
MODULE 3	8**Hrs**
AGILE PROJECT MANAGEMENT: Sprint Planning, Sprint Reviews, Sprint Retrospectives, Sprint Planning - Agile release and iteration (sprint) planning, Develop Epics and Stories, Estimating Stories, Prioritizing Stories (WSJF technique from SAFe), Iterations/Sprints Overview. Velocity Determination, Iteration Planning Meeting, Iteration, Planning Guidelines, Development, Testing, Daily Stand-up Meetings, Progress Tracking, Velocity Tracking, Monitoring and Controlling: Burn down Charts, Inspect & Adapt (Fishbone Model), Agile Release Train	
MODULE 4	**7Hrs**
AGILE TESTING : Testing: Functionality Testing, UI Testing(Junit, Sonar), Performance Testing, Security Testing, A/B testing; Agile Testing: Principles of agile testers; The agile testing quadrants, Agile automation, Test automation pyramid; Test Automation Tools - Selenium, Traceability matrix;	
MODULE 5	**8Hrs**
DEVOPS: DevOps: Continuous Integration and Continuous Delivery; CI/CD: Jenkins, Git/Github Creating pipelines, Setting up runners Containers and container orchestration (Dockers and Kubernetes) for application development and deployment; Build tools - maven; Checking build status; Configuration management - puppet, chef, ansible; Fully Automated Deployment; CM - Continuous monitoring with Nagios; Introduction to DevOps on Cloud	

List of Laboratory/Practical Experiments activities to be conducted:
1. Setting up Devops Environment
2. Writing Requirements Document, Requirement Analysis (user stories)
3. Estimation and Scrum Planning
4. Implementation and Testing Using Iterative Sprint Model
5. Test Automation using Selenium
6. Unit Testing using Junit or Sonar or Python Test framework
7. CI/CD using Jenkins as Orchestrion platform
8. Containerzation using Docker or Kubernetes

TEXT BOOKS:

1. Essential Scrum: A Practical Guide to the Most Popular Agile Process Kenneth S.Rubin 2012, published by Addison-Wesley Professional
2. Agile Software Development: The Cooperative Game Alistair Cockburn 2nd Edition, 2006, Addison-Wesley Professional

REFERENCES:

1. Scrum and XP from the Trenches Henrik Kniberg 2nd Edition, 2015, Published by C4Media, publisher of InfoQ.com
2. Agile Project Management: Creating Innovative Products, Second Edition By Jim Highsmith, Addison-Wesley Professional, 2009
3. Agile Project Management: Managing for Success, By James A. Crowder, Shelli Friess,Springer 2014

SEMESTER	**V**					
YEAR	**III**					
COURSE CODE	21CS3512					
TITLE OF THE COURSE	**DATA WARE HOUSE AND DATA MINING**					
SCHEME OF INSTRUCTION	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	3	**-**	**-**	**-**	**39**	**3**

Perquisite Courses (if any)			
#	Sem/Year	Course Code	Title of the Course
*	*	**	***

COURSE OBJECTIVES:

- To extract knowledge from data repository for data analysis,
- Apply preprocessing statistical methods for any given raw data.
- Characterize the kinds of patterns that can be discovered by association rule mining, classification and clustering.
- Master data mining techniques in various applications like social, scientific and environmental context.
- Develop skill in selecting the appropriate data mining algorithm for solving practical problems.

COURSE OUTCOMES:

CO No.	Outcomes	Bloom's Taxonomy Level
CO1	Examine the techniques of data warehousing like Building the warehouse, mapping the Data Warehouse to Multiprocessor Architecture	L4
CO2	Apply the association rules like APRIORI, FP Growth Algorithm and Correlation Analysis for mining the data	L3
CO3	Design the classification algorithms like Decision Tree Induction, Bayesian Classification, Rule Based Classification, Support Vector Machines, K-nearest neighbor classification to classify the data	L6
CO4	Develop the basic data mining algorithms like Model Based clustering algorithms, Grid Based Methods, Density Methods and Constraint Based Clustering Analysis	L3
CO5	Evaluate various mining techniques on complex data objects, Partition Algorithms, Support and Confidence Measures	L5

COURSE CONTENT:	
71	
MODULE 1: DATA WAREHOUSING	**7Hrs**
Data Warehouse, Data warehousing Components –Building a Data warehouse - Mapping the Data Warehouse to a Multiprocessor Architecture – DBMS Schemas for Decision Support – Data Extraction, Cleanup, Transformation and loading, Tools, Metadata.	

MODULE 2: BUSINESS ANALYSIS	7Hrs
Reporting and Query tools and Applications – Tool Categories – The Need for Applications – Cognos Impromptu – Online Analytical Processing (OLAP) – OLAP Guidelines – Multidimensional Data Model - Multidimensional versus Multi-relational OLAP – OLAP Tools and the Internet. Case study: Data Warehouse tools in cloud (MS Azure, AWS)	
MODULE 3: DATA MINING	**9 Hrs**
Introduction to Data – Types of Data – Types of Data-attributes and measurements - types of data sets, Data Quality - Data Mining Functionalities – Interestingness of Patterns – Classification of Data Mining Systems, Data Mining Task Primitives – Integration of a Data Mining System with a Data Warehouse – Issues in DM, KDD process. Data Preprocessing.	
MODULE 4: CLUSTERING AND TRENDS IN DATA MINING	**8Hrs**
Cluster Analysis, Categorization of Major Clustering Methods - Partitioning Methods: K-means clustering. Hierarchical Methods: Agglomerative Methods and Divisive Methods, Density Based Methods, Grid Based Methods, Model Based Clustering Methods, Clustering High Dimensional Data, Constraint Based Cluster Analysis, Outlier Analysis, Data Mining Applications.	
MODULE 5: ASSOCIATION RULE MINING AND CLASSIFICATION	**8 Hrs**
Frequent Item Set Generation, The APRIORI Principle, Support and Confidence Measures, Association Rule Generation, APRIORI Algorithm, Correlation Analysis. Classification and Prediction: General Approaches to solving a classification problem, Evaluation of Classifiers, Classification techniques	

TEXT BOOKS:

1. Alex Berson and Stephen J Smith, "Data Warehousing, Data Mining & OLAP", Tata McGraw – Hill Edition, Tenth Reprint, 2007
2. Jiawei Han and Micheline Kamber, "Data Mining Concepts and Techniques", Second Edition, Elsevier, 2007

REFERENCES:

1. Pang-Ning Tan, Michael Steinbach and Vipin Kumar, "Introduction to Data Mining", Person Education, 2007.
2. K.P. Soman, Shyam Diwakar and V. Ajay ", Insight into Data mining Theory and Practice", Easter Economy, Edition, Prentice Hall of India, 2006.
3. G. K. Gupta, "Introduction to Data Mining with Case Studies", Easter Economy Edition, Prentice Hall of India, 2006
4. Daniel T. Larose, "Data Mining Methods and Models", Wile-Interscience, 2006

SEMESTER	V
YEAR	III
COURSE CODE	21CS3513
TITLE OF THE COURSE	MOBILE COMPUTING AND APPS DEVELOPMENT

SCHEME OF INSTRUCTION	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	3	-	-	-	39	3

Perquisite Courses (if any)			
#	Sem/Year	Course Code	Title of the Course
*	**	**	***

COURSE OBJECTIVES:

- To understand the basic concepts of mobile computing
- To learn the setup of Android development environment
- To illustrate the interaction of app with the user interface and handling various activities
- To identify the options for saving the persistent application data
- To gain knowledge about different mobile platforms and application development

COURSE OUTCOMES:

CO No.	Outcomes	Bloom's Taxonomy Level
CO1	Create, test and debug Android application by setting up the Android development environment	L6
CO2	Implement adaptive and responsive user interfaces that work across various devices	L2
CO3	Demonstrate the techniques involved to store, share andretrieve data in Android applications	L3
CO4	Acquire technical competency and skills in developing applications using Android and cross-platform	L3

COURSE CONTENT:	
MODULE 1: INTRODUCTION TO MOBILE COMPUTING	7Hrs
Introduction to mobile computing, Architecture of mobile network, Generations of mobile communication, mobile operating systems, Application of mobile communication, Challenges of mobile communication.	
MODULE 2	8Hrs
Introduction, trends, platforms, Android Development Setup like, Android Studio, Eclipse,	

Android SDK, tools. Emulator setup. App behavior on the Android Runtime (ART). Platform Architecture. Application framework and basic App Components resources. HelloWorld program in Android Studio
MODULE 3: MOBILE APP DEVELOPMENT USING ANDROID 9Hrs

MOBILE APP DEVELOPMENT USING ANDROID: Android user Interface – Layouts (Linear, Absolute, Table, Relative, Frame and Scroll), values, asset XML representation, generate R.Java file, Android manifest file.Activities, Intent and UI Design - activities life-cycle. Android Components – layouts,fragments, basic views (Button, Edit Text, Check box, Toggle Button, Radio Button), list views, picker views, adapter views, Spinner views, Menu, Action Bar and Managing data using SQLite database (Database create, Read, Update and delete).	
MODULE 4 : MESSAGING AND LOCATION BASED SERVICES	8Hrs
Sending SMS and mail, Google Maps – Displaying Google Maps in Andriod application, Networking – How to connect to Web using HTTP, Publishing Android Applications – howto prepare application for deployment, exporting application as an APK file and signing it with new certificate, how to distribute new android application and publish android application on market place	
MODULE 5: DATA PERSISTENCE AND GOOGLE APIS FOR ANDROID:	7Hrs
Introduction of Google APIs for Android. SQLite Databases. CROSS-PLATFORM APP DEVELOPMENT - Introduction to Cross platform App Development - Difference to nativeapps, Pros and cons, Development tools.	

TEXT BOOKS:

1. Mobile Cloud Computing by Debashis De, CRC Press, Taylor & FrancisGroup
2. **Head First Android Development by Jonathan Simon O'reilly Publications**

REFERENCES:

1. **Learning Android by Marko Gargenta O'reilly Publications**
2. Jochen H. Schller, "Mobile Communications", Second Edition, Pearson Education,New Delhi, 2007.
3. Uwe Hansmann, LotharMerk, Martin S. Nicklons and Thomas Stober, "Principlesof Mobile Computing", Springer, 2003.
4. Erik Hellman, "Android Programming – Pushing the Limits", 1st Edition, WileyIndia Pvt Ltd, 2014
5. J F DiMarzio, "Beginning Android Programming with Android Studio", 4thEdition, Wiley India Pvt Ltd, 2016.

SEMESTER	V					
YEAR	III					
COURSE CODE	21CS3514					
TITLE OF THE COURSE	MOOC					
SCHEME OF Instruction	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	-	--	-	--	39	3

Course Outcomes:

1. Enabling students to obtain certificates to make students employable in the industry or pursue a higher education program.
2. Relevant exposure to tools and technologies are being offered.

Massive Open Online Courses (MOOCs) – Guidelines & Policy

1. Students shall enroll the MOOC courses that is available on the NPTEL/SWAYAM (Swayam.gov.in) platform whenever it notifies (twice in a year).
2. The list of NPTEL / SWAYAM courses related to Computer Science & Engineering thatis in line with the students interest will be announced at the departmental level for enrollment.
    - That is, the predefined list of courses is provided by the department to thestudents, and only those courses shall be considered and not others.
3. Students shall also enroll in Coursera / Udemy / Udacity / Infosys Spring Board, where DSU can consider the grades / marks provided by these platforms if they are proctored ones. Examinations are to be conducted by DSU if proctored assessments are not conducted by these platforms.
4. The MOOCs courses option shall be considered only for students having a minimum CGPA of **6.75.**
5. The interested student has to enroll as per the guidelines of the NPTEL / SWAYAM or other platforms mentioned in item 3 within enrollment end date.
6. The credits assigned would depend on the number of weeks. The department shallconsider 12 weeks course to map for 03 Credits.
7. A faculty member shall be appointed as SPOC to keep a track of students undertaking courses and collect certificates from students upon completion on theplatforms mentioned above.
8. Student has to pursue and acquire a certificate for a MOOCs course and after successful completion, the student shall submit the certificate to the Department and credits shall be transferred to the grade card accordingly based on the items 1-3 above.
9. The examination fee for obtaining the certificate shall be borne by the student.

10. In case a student fails to complete the MOOC course, then the student shall repeatthe same on the NPTEL/SWAYAM or other platforms mentioned in item 3 or the student may opt for department elective with permission of the department chair.
11. Following is the proposed range for the award of grades towards the creditstransfer.

Range: Consolidated MOOC Score (Assignment+ Proctored exam)	Proposed Grade Point	Grade
90-100	10	O
80-89	9	A+
70-79	8	A
60-69	7	B+
55-59	6	B
50-54	5	C
40-49	4	P
Less than 40	0	F

SEMESTER	**VI**					
YEAR	**III**					
COURSE CODE	**21CS3601**					
TITLE OF THE COURSE	**COMPILER DESIGN AND SYSTEMS SOFTWARE**					
SCHEME OF INSTRUCTION	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	3	**1**	**-**	**-**	**39**	**4**

Perquisite Courses (if any)			
#	Sem/Year	Course Code	Title of the Course
*	**	**	***

COURSE OBJECTIVES:

1. To explain the basic system software components such as assembler, loader, linkers, compilers.
2. Provide an understanding of the fundamental principles in compiler design
3. To discuss the techniques of scanning, parsing & semantic elaboration well enough to build or modify front end.
4. To illustrate the various optimization techniques for designing various optimizing compilers.

COURSE OUTCOMES:

CO No.	Outcomes	Bloom's Taxonomy Level
CO1	Identify the data structures, algorithm, machine dependent Assembler features and build the object code for Simplified Instructional Computer program	L2
CO2	Infer how linker and loader builds an executable program from an object module generated by assembler	L4
CO3	Interpret the major phases of compilation and to apply the knowledge of Lex tool & YACC tool to build the appropriate parsing application	L2
CO4	Compare and Contrast various top down and bottom up parsing techniques to analyze grammatical structures involved in compiler construction.	L2
CO5	Use formal attributed grammars for specifying the syntax and semantics of programming languages.	L3
CO6	Select various optimization techniques used for dataflow analysis and build machine code from the source code of a novel language.	L2

COURSE CONTENT:	
MODULE 1: Introduction to System Software, ASSEMBLERS	**9Hrs**
Introduction to System Software, Machine Architecture of SIC and SIC/XE. ASSEMBLERS: Basic assembler functions: A simple assembler, Assembler algorithm and data	

structures, Machine dependent assembler features: Instruction formats and addressing modes – Program relocation, Machine independent assembler features: Literals, Symbol-defining statements, Expressions, Program blocks
MODULE 2 : LOADERS AND LINKERS: **7 Hrs**
Basic loader functions: Design of an Absolute Loader, A Simple Bootstrap Loader, Machine dependent loader features: Relocation, Program Linking, Algorithm and Data Structures for Linking Loader, Machine-independent loader features: Automatic Library Search, Loader Options, Loader design options: Linkage Editors, Dynamic Linking
MODULE 3: COMPILERS **8Hrs**
Introduction: Language Processors, Structure of compiler, The science of building a compiler, Applications of compiler technology. LEXICAL AND SYNTAX ANALYSIS: Role of lexical Analyzer, Specification of Tokens, Lexical Analyzer generator Lex. SYNTAX ANALYSIS I: Role of Parser, Syntax error handling, Error recovery strategies, Writing a grammar: Lexical vs Syntactic Analysis, Eliminating ambiguity, Left recursion, Left factoring.
MODULE 4: SYNTAX ANALYSIS II **8 Hrs**
Top down parsing: Recursive Descent Parsing, First and follow, LL (1), –Bottom up parsing: Shift Reduce Parsing, Introduction to LR parsing Simple LR: Why LR Parsers, Items and LR0 Automaton, The LR Parsing Algorithm. SYNTAX-DIRECTED TRANSLATION: Syntax-Directed Definitions: Inherited and Synthesized Attributes, Evaluation orders for SDDs: Dependency graphs, Ordering the evaluation of Attributes, S-Attributed Definition, L-Attributed Definition, Application: Construction of Syntax Trees.
MODULE 5: INTERMEDIATE CODE GENERATION **7 Hrs**
Three Address Code: Addresses and Instructions, Quadruples, Triples, indirect triples. CODE GENERATION: Issues in the design of code generator, Basic Blocks, Optimization of Basic Blocks, The Code Generation Algorithm, Peephole optimization. MACHINE INDEPENDENT OPTIMIZATION: The Principal Sources of Optimization

TEXT BOOKS:

1. Leland L. Beck, "System Software – An Introduction to Systems Programming", 3rd Edition, Pearson Education Asia, 2006.
2. Alfred V Aho, Monica S. Lam, Ravi Sethi and Jeffrey D Ullman, "Compilers – Principles, Techniques and Tools", 2nd Edition, Pearson Education, 2007.

REFERENCES:

1. V. Raghavan, Principles of Compiler Design‖, Tata McGraw Hill Education Publishers, 2010.
2. Keith D Cooper and Linda Torczon, Engineering a Compiler, Morgan Kaufmann Publishers Elsevier Science, 2004.
3. D.M.Dhamdhere, Systems Programming and operating systems, Second Revised edition, Tata McGraw Hill.

SEMESTER	**VI**					
YEAR	**III**					
COURSE CODE	**21CS3602**					
TITLE OF THE COURSE	**COMPUTER NETWORKS**					
SCHEME OF INSTRUCTION	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	3	**-**	**-**	**-**	**39**	**3**

Perquisite Courses (if any)			
#	Sem /Year	Course Code	Title of the Course
***	***	***	***

COURSE OBJECTIVES:

- To introduce the fundamental aspects of various types of computer networks.
- To demonstrate the TCP/IP and OSI models with merits and demerits.
- To Understand the working principle of layering structure and basic network components
- To explore the features of each layer by various approach and methods

COURSE OUTCOMES:

CO No.	Outcomes	Bloom's Taxonomy Level
CO1	Identify and compare among different layers of networking and associated components.	L3
CO2	Implement error control and error detection mechanisms (CRC, Hamming Code) using the concept of the data link layer.	L3
CO3	Differentiate IP addressing modes, implement routing algorithms, and determine the range of congestion in any network.	L3
CO4	Identify the issues of the Transport layer to analyze the congestion control mechanism	L3
CO5	Compare application layer protocols (WEB and HTTP, FTP, E-MAIL (SMTP, POP3), TELNET, DNS, SNMP).	L4

COURSE CONTENT	
MODULE 1: Overview of Networks	**9 Hrs**
Network Components- Network Physical Structure, Classification of networks (LAN-MAN-WAN), Protocols and Standards, Data representation and data flow, Layered Architecture – Comparison of the OSI and TCP/IP reference model. Physical Layer: Introduction to wired and wireless transmission media. Transmission mode (Serial/Parallel signals, Analog/Digital Signals and Periodic/Aperiodic Signals), Line coding	

Schemes.	
MODULE 2: Data Link Layer	9 **Hrs**
Data Link Layer – MAC (Media Access Control) and LLC (Logical Link Control) sublayer Functionalities– Design Issues: Framing – Flow control (Simplest protocol, Stop and wait, sliding window) – Error control (CRC, Hamming code) — Ethernet Basics-Multi Access Protocols: ALOHA, CSMA/CD, Connecting Devices: Hubs, Bridges, Switches, Routers, and Gateways	
MODULE 3: Network Layer	**8 Hrs**
Network Layer Design issues, Routing Protocol Basics, Routing Algorithm (Distance Vector Routing, Link State Routing and Hierarchical Routing). IP addressing, IP Packet format IPV4, IPV6 and IP Tunneling. Congestion control algorithms, QoS (Traffic Shaping, Packet Scheduling).	
MODULE 4: Transport Layer	7 **Hrs**
Transport Layer functions- Multiplexing and Demultiplexing. Introduction to TCP and UDP, The TCP Service Model, The TCP Segment Header, The TCP Connection Management, TCP Flow Control- Sliding Window, TCP Congestion Control, User Datagram Protocol	
MODULE 5: Application Layer	6 **Hrs**
Principles of Network Applications, WEB and HTTP, FTP, E-MAIL (SMTP, POP3), TELNET, DNS, SNMP	

TEXT BOOKS:

1. Behrouz A. Forouzan, Data Communications and Networking, Fifth Edition TMH, 2013.
2. Computer Networks - Andrew S Tanenbaum, 5th Edition, Pearson Education.

REFERENCES:

1. James F. Kurose, Keith W. Ross, “Computer Networking: A Top-Down Approach”, Seventh Edition, Pearson Education, 2017.
2. Larry L. Peterson, Bruce S. Davie, “Computer Networks: A Systems Approach”, Fifth Edition, Morgan Kaufmann Publishers Inc., 2011.
3. William Stallings, “Data and Computer Communications”, Tenth Edition, Pearson Education, 2014.

SEMESTER	VI					
YEAR	III					
COURSE CODE	21CS3603					
TITLE OF THE COURSE	CLOUD COMPUTING					
SCHEME OF INSTRUCTION	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	3	-	-	-	39	3

Perquisite Courses (if any)			
#	Sem/Year	Course Code	Title of the Course
***	***	***	***

COURSE OBJECTIVES:

- Understand various basic concepts related to cloud computing technologies
- Understand the architecture and concept of different cloud models: IaaS, PaaS, SaaS
- Understand the applications of Cloud Computing
- Get exposure to Microsoft Azure, Google Cloud Platform, Amazon Web Services

COURSE OUTCOMES:

CO No.	Outcomes	Bloom's Taxonomy Level
CO1	Illustrate the main concepts, models, strengths, and limitations of cloud computing and make use of NIST cloud computing architecture to solve architecture design challenges	L3
CO2	Outline the key technologies and approaches for implementation of the cloud and also analyze the authentication, confidentiality, and privacy issues in cloud computing	L4
CO3	Evaluate the relative advantages and disadvantages of Virtualization technology and the taxonomy of Virtualization	L5
CO4	Infer the key and enabling technologies that help in the development of the cloud and the steps involved in migration to the cloud	L2
CO5	Utilize the main dimensions of current cloud platforms and identify the appropriate cloud services for a given application	L6

COURSE CONTENT:	
MODULE 1: INTRODUCTION	8Hrs

Basics of cloud computing, Cloud Computing Models (Paas, Saas, Iaas), Understanding Public Clouds, Private Clouds, Community Cloud and Hybrid Clouds, Cloud Computing Benefits and risks, Cloud Computing Challenges, Cloud Computing Architecture and Virtualization	
MODULE 2: CLOUD Technologies	7 Hrs
Overview of Cloud Computing techniques (Grid Computing,Cloud Computing, Utility Computing, Fog Computing, Edge computing), Introduction to Cloud security.	
MODULE 3: CLOUD VIRTUALIZATION TECHNOLOGY	8Hrs
Introduction, why virtualization, virtualization benefits, Types of Virtualization-Storage, Application & Network Virtualization, implementing virtualization, Hypervisor.	
MODULE 4: ACCESSING THE CLOUD AND MIGRATING TO THE CLOUD	8Hrs
Accessing the Cloud: Cloud Web access technologies (SOAP, REST), Platforms, Web applications framework, web hosting service, web APIs, web browsers. Migrating to the Cloud: Broad approaches to migrating into the cloud, the seven-step model of migration in to a cloud.	
MODULE 5: CLOUD APPLICATIONS	8Hrs
Cloud Platforms in Industry: Amazon Web Services, Google Cloud Platform, Microsoft Azure. Cloud Applications: Scientific Applications (Healthcare: ECG Analysis in the Cloud) and Business and Consumer Applications (Social Networking, Smart Grids)	

TEXT BOOKS:

1. **Rajkumar Buyya, Christian Vecchiola, and Thamarai Selvi Mastering Cloud. Computing McGraw Hill Education**
2. **Cloud Computing, Dr. Kumar Saurabh, Wiley Publications, 2012**

REFERENCES:

1. Guide to Cloud Computing, Richard hill, Springer Publications, 2013
2. **Cloud Computing A Practical Approach, Anthony T Velte et.al, MC Graw Hill publications, 2014**
3. Cloud Computing Principles and Paradigms, Rajkumar Buyya et.al, WileyPublications, 2015
4. **Cloud Computing Technologies and Strategies of the Ubiquitous data center, Brain J.S et.al, CRC Press, 2014**

SEMESTER	**VI**					
YEAR	**III**					
COURSE CODE	**21CS3604**					
TITLE OF THE COURSE	**COMPILER DESIGN AND SYSTEM SOFTWARE LAB**					
SCHEME OF INSTRUCTION	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	-	-	**2**	**-**	**26**	**1**

Perquisite Courses (if any)			
#	Sem/Year	Course Code	Title of the Course
***	***	***	***

COURSE OBJECTIVES:

- Experiment on the basic techniques of compiler construction and tools that can used to perform syntax-directed translation of a high-level programming language into an executable code.
- Know the implementation of assemblers, loaders and various parsing techniques.
- Learn how to optimize and effectively generate machine codes.

COURSE OUTCOMES:

CO No.	**Outcomes**	**Bloom's Taxonomy Level**
CO1	ldentify patterns, tokens & regular expressions for lexical analysis.	L2
CO2	Develop LEX and YACC programs for lexical and syntax analysis phases of Compiler.	L3
CO3	Implement the two-pass assembler and absolute loader to translate assembly language into machine code and load the machine code into RAM for execution	L3
CO4	Implement the bottom-up parsing applied in the syntax analysis phase of the compiler	L3
CO5	Develop first sets of Context free grammar to generate a predictive parser used to check whether the input source code follows the syntax of the programming language.	L3

List of Laboratory/Practical Experiments activities to be conducted
1a. Program to count the number of characters, words, spaces and lines in a given input file.
1b. Program to recognize and count the number of identifiers in a file.

2a. Program to count the numbers of comment lines in a given C program. Also eliminate them and copy the resulting program into separate file.
2b. Program to recognize whether a given sentence is simple or compound.
3a. Program to count no of: i.+ve and –ve integers ii. +ve and –ve fractions
3b. Program to count the no of „scanf‟ and „printf‟ statements in a C program. Replace them with “readf‟ and “writef‟ statements respectively.
4.Program to evaluate arithmetic expression involving operators +,-,*,/
5. Program to recognize a valid variable which starts with a letter, followed by any number of letters or digits.
6. Program to recognize the strings using the grammar ($a^n b^n$;n>=0)
7. C Program to implement Pass1 of Assembler
8. C Program to implement Absolute Loader
9. C program to find the FIRST in context free grammar.
10.C Program to implement Shift Reduce Parser for the given grammar E →E+E E→E*E E→(E) E → id
11. C Program to implement intermediate code generation for simple expression

TEXT BOOKS:

1. Leland L. Beck, “System Software – An Introduction to Systems Programming”, 3rd Edition, Pearson Education Asia, 2006.
2. Alfred V Aho, Monica S. Lam, Ravi Sethi and Jeffrey D Ullman, “Compilers – Principles, Techniques and Tools”, 2nd Edition, Pearson Education, 2007.

SEMESTER	VI					
YEAR	III					
COURSE CODE	21CS3605					
TITLE OF THE COURSE	COMPUTER NETWORKS LAB					
SCHEME OF INSTRUCTION	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	-	-	**2**	**-**	**26**	**1**

COURSE CONTENT

List of Laboratory/Practical Experiments activities to be conducted

PART A

1. Implement three nodes point – to – point network with duplex links between them. Set the queue size, vary the bandwidth and find the number of packets dropped.
2. Implement transmission of ping messages/trace route over a network topology consisting of 6 nodes and find the number of packets dropped due to congestion.
3. Implement an Ethernet LAN using n nodes and set multiple traffic nodes and plot congestion window for different source / destination.
4. Implement simple ESS and with transmitting nodes in wire-less LAN by simulation and determine the performance with respect to transmission of packets.
5. Implement and study the performance of GSM on NS2/NS3 (Using MAC layer) or equivalent Environment.
6. Implement and study the performance of CDMA on NS2/NS3 (Using stack called Call net) or equivalent environment.

PART B

Implement the following in Java:

7. Write a program for error detecting code using CRC.
8. Write a program to find the shortest path between vertices using bellman-ford algorithm.
9. Using TCP/IP sockets, write a client – server program to make the client send the file name and to make the server send back the contents of the requested file if present. Implement the above program using as message queues or FIFOs as IPC channels.
10. Write a program on datagram socket for client/server to display the messages on client side, typed at the server side.
11. Write a program for simple RSA algorithm to encrypt and decrypt the data.
12. Write a program for congestion control using a leaky bucket algorithm.

Perquisite Courses (if any)			
#	**Sem/Year**	**Course Code**	**Title of the Course**
***	***	***	***

SEMESTER	**VI**					
YEAR	**III**					
COURSE CODE	**21CS3606**					
TITLE OF THE COURSE	**BLOCKCHAIN AND DISTRIBUTED LEDGER**					
SCHEME OF INSTRUCTION	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	3	**-**	**-**	**-**	**39**	**3**

COURSE OBJECTIVES:

- Learn the underlying principles and techniques associated with block chain Technologies.
- Understand and describe how blockchain works
- Familiarize with Ethereum, smart contracts and related technologies, and solidity language.
- Understand the application of Blockchain in various domains

COURSE OUTCOMES:

CO No.	Outcomes	Blooms Taxonomy Level
CO1	Outline the basic concepts of blockchain and cryptography ,mining ,merkle tree concepts used in blockchain to develop decentralized applications	L3
CO2	Use solidity programming for smart contract development in real world applications such as library management system ,student management system ,employee management system.	L3
CO3	Implement Ethereum blockchain applications using geth, metamask, ganache , truffle blockchain tools.	L3

CO4	Develop Block chain Application for IoT smart home ,healthcare using hyperledger platform	L6
CO5	Adapt the advanced concepts of blockchain programming language and tools to develop complex blockchain application	L6

COURSE CONTENT:	
MODULE 1: Introduction to Blockchain	8Hrs
Distributed systems, P2P network Architecture of Blockchain, Generic elements of a blockchain: How blockchain works, Benefits, features, and limitations of blockchain How blockchain accumulates blocks, types of blockchain, Distributed ledger, Consensus mechanisms-Proof of work, Proof of Stake, Proof of Authority, CAP theorem, Decentralization, Disintermediation, Ecosystem - Storage, Communication and Computation	
MODULE 2: Cryptography and Smart Contracts	8Hrs
Symmetric cryptography (DES, AES), Asymmetric cryptography, Public and Private keys, Algorithms - RSA, Hash functions, SHA, SHA-256 Smart contracts - Benefits of Smart contracts, Solidity Programming-Types, Literals, Enums, write basic program using Solidity, Compile, verify and deploy.	
MODULE 3: Ethereum Blockchain	8Hrs
The Ethereum network, Ethereum Virtual Machine Execution Environment, Opcodes and their meaning, Structure of a Block, Genesis Block, Merkle tree, Geth, Transactions, Transaction receipts, Nonce, Gas - gasPrice, gasLimit, Ether, Mining, Wallets, Ethereum network (main net, test net), Metamask	
MODULE 4: Ethereum Development	8Hrs
Infura, Web3.0 for Blockchain, Web3J -Java frontend, Creating Blockchain network and peering, Truffle - build contract, migrate and deploy, Ganache CLI	
MODULE 5: Hyperledger	7Hrs
Projects under Hyperledger, Hyperledger reference architecture, Hyperledger design principles, Hyperledger Fabric, Hyperledger Sawtooth, Case study: Blockchain in IoT	

TEXT BOOKS:

1. Mastering Blockchain, Third Edition, Published by Packt Publishing Ltd, Published 2020, Imran Bashir
2. Solidity Programming Essentials, First Edition, Published by Packt Publishing Ltd, April 2018 Blockchain for Dummies, Manav Gupta, IBM Limited Edition, John Wiley & Sons, Inc. 2017

SEMESTER	VI					
YEAR	III					
COURSE CODE	21CS3607					
TITLE OF THE COURSE	MACHINE LEARNING FOR HEALTHCARE					
SCHEME OF INSTRUCTION	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	3	-	-	-	39	3

Perquisite Courses (if any)			
#	Sem/Year	Course Code	Title of the Course
*	*	**	***

COURSE OBJECTIVES:

- To introduce the students to healthcare domain and to make them understandand practice to use machine learning techniques to data in the healthcare domain

COURSE OUTCOMES:

CO No.	Outcomes	Bloom's Taxonomy Level
CO1	Identify different problems in the healthcare industry that need a Machine Learning solution and Analyze data to develop predictive models for risk stratification, disease diagnosis, prognosis, treatment response prediction, and anomaly detection.	L4
CO2	Recognize the challenges associated with medical image modalities and clinical text including privacy, data quality, handling missing values, use data preprocessing techniques to resolve them.	L2
CO3	Create Automatic workflow, applying deep learning algorithms which are more appropriate for medical imaging and wearable sensor of healthcare applications using Python-based toolkits	L6
CO4	Develop healthcare applications using Python-based toolkits incorporating NLP algorithms and machine learning techniques specifically tailored for clinical text and electronic health record (HER) data.	L4

COURSE CONTENT:		
MODULE 1	88	8Hrs

Knowing Healthcare Industry: Overview of Healthcare & Life science Industry, Introduction to healthcare informatics, Key Components in Health care, Health Level Seven, Medical Standards and Coding Types, GlobalHealthcare Challenges and Trends; Past-Present-Future of AI&ML in Healthcare.	
MODULE 2	**9Hrs**
Advanced Analytics in Health Care: Overview of clinical care, Clinical Data, Data Types; Risk Stratification; Survival Modelling; Disease progression Modelling, Causal Inference, Re-enforcement learning in healthcare applications	
MODULE 3	**8Hrs**
Medical Image Diagnostics and NLP for healthcare: Medical Image modalities and management; ML applications in medical Ology space (cardiology, ophthalmology, dermatology, pathology, oncology, haematology, odontology, osteology, pulmonology); NLP for Healthcare: Payer Analytics – Insurance	
MODULE 4	**8Hrs**
Precision Medicine, Automating clinical workflow, Regulation of AI/ML, the challenge in deploying ML model, Public Health – Government, Provider Analytics, Care Management System, Wearabledevices and Medical Bots.	
MODULE 5	**6Hrs**
Applications of Machine learning models (Linear regression, SVM, Random forest.) and Deep learning models (CNN, RNN….) for the Healthcare area (Case study)	

TEXT BOOKS:

1. Sumeet Dua, U. Rajendra Acharya, Prerna Dua (Editors), Machine Learning in Healthcare Informatics, Intelligent Systems Reference Library 56, Springer,
2. Sergio Consoli, Diego ReforgiatoRecupero, Milan Petkovic (Editors), Data Science for Healthcare Methodologies and Applications

REFERENCES:

1. Thomas M. Deserno, Fundamentals of Bio-Medical Image processing, Biological and Medical Physics, Biomedical Engineering, Springer, ISBN 978-3-642-15816-2, 2011

SEMESTER	VI					
YEAR	III					
COURSE CODE	21CS3608					
TITLE OF THE COURSE	DEEP LEARNING					
SCHEME OF INSTRUCTION	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/ProjectsHours	Total Hours	Credits
	3	-	-	-	39	3

Perquisite Courses (if any)			
#	Sem/Year	Course Code	Title of the Course
***	***	***	***

COURSE OBJECTIVES:

- To understand the basic building blocks and general principles that allows one to design Deep learning algorithms
- To become familiar with specific, widely used Deep learning networks
- To introduce building blocks of Convolution neural network architecture
- To learn to use deep learning tools and framework for solving real-life problems

COURSE OUTCOMES:

CO No.	Outcomes	Bloom's Taxonomy Level
CO1	Identify the basic building blocks and general principles that allows one to design Deep learning algorithms	L2
CO2	Determine the working of various Deep learning networks	L3
CO3	Design and Implement Convolution neural network models for image processing applications	L3
CO4	Develop solutions for real-life problems using deep learning tools and frameworks	L3

COURSE CONTENT:	
MODULE 1: Introduction to Deep Learning	8Hrs
Introduction to Neural Networks: Single layer and Multilayer NN, training neural networks, activation functions, loss functions, Model Selection. Introduction to Deep Learning, Principles of Deep Networks and Building blocks of deep networks.	

MODULE 2	7Hrs
Mathematical background for Deep learning- Data Manipulation and Data Preprocessing, Linear Algebra,Calculus, Probability.	
MODULE 3	**8Hrs**
Forward Propagation, Backward Propagation, and Computational Graphs Layers and Blocks, shallow neural network, deep neural network, Optimization for Training Deep Models.	
MODULE 4	**8 Hrs**
Convolutional Neural Networks (CNNs) - Biological inspiration, Mapping of Human Visual System and CNN. Convolution operation, Convolutional Layers, Padding and Stride, Batch normalization and layers, Subsampling, Pooling.	
MODULE 5	**8Hrs**
Unsupervised Pretrained Networks (UPNs)- Autoencoders, Deep Belief Networks (DBNs) Introduction to Generative Adversarial Networks (GANs), Deep Learning Applications in Healthcare and other areas (Case study)	

TEXT BOOKS:

1. Aston Zhang, Zack C. Lipton, Mu Li, Alex J. Smola, “Dive into Deep Learning”, Amazon Science, 2020
2. Josh Patterson and Adan Gibson, “Deep Learning a Practitioners Approach”, July, 2018.

REFERENCES:

1 Tom Mitchell, Machine Learning, McGraw-Hill, 1997

2. Ian Goodfellow, Yoshua Bengio, Aaron Courville, “Deep Learning”, The MIT Press, 2016

3. François Chollet, “Deep Learning Python”, Manning Publications, 2018

4. Aurélien Géron, “Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, andTechniques to Build Intelligent Systems”, O'Reilly Media; 1 edition (April 9, 2017)

5 “Neural Networks: A Comprehensive Foundation,”S. Haykin, 2ndEd, Prentice Hall of India, 2003.

SEMESTER	**VI**					
YEAR	**III**					
COURSE CODE	**21CS3609**					
TITLE OF THE COURSE	**DIGITAL IMAGE PROCESSING**					
SCHEME OF Instruction	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	3	**-**	**-**	**-**	**39**	**3**

Perquisite Courses (if any)			
#	Sem/Year	Course Code	Title of the Course
***	***	***	***

COURSE OBJECTIVES:

- To understand and to become familiar with the fundamentals of Digital Image Processing.
- To get exposed to simple image enhancement techniques in Spatial and Frequency domain.
- To learn concepts of degradation function and restoration techniques.
- To study the image segmentation and representation techniques.

COURSE OUTCOMES:

CO No.	Outcomes	Bloom's Taxonomy Level
CO1	Use the basics and fundamentals of digital image processing, such as digitization, sampling, quantization, and 2D-transforms for image segmentation, object detection and recognition.	L3
CO2	Preprocess the images using the techniques of smoothing, sharpening and enhancement in spatial domain and frequency domain for object detection and recognition.	L4
CO3	Use image degradation, restoration and morphological processing to enhance preprocessed images.	L4
CO4	Apply object detection and recognition techniques using point, line, edge detection and thresholding.	L3
CO5	Apply the concept of image segmentation using Clustering, Graph Cuts, Morphological Watersheds.	L5

COURSE CONTENT:	
MODULE 1 : INTRODUCTION	**8 Hrs**
Overview of Digital Image Processing, Origins of Digital Image Processing, Examples of fields that use Digital Image Processing, Fundamental Steps in Digital Image Processing, Components in Digital Image Processing System. **DIGITAL IMAGE FUNDAMENTALS**	

Elements of Visual Perception, Image Sensing and Acquisition, Image Sampling and Quantization, Some Basic Relationships Between Pixels.	
MODULE 2 : IMAGE ENHANCEMENT IN THE SPATIAL DOMAIN	**8 Hrs**
Basic Intensity Transformation Functions, Histogram Processing, Fundamentals of Spatial Filtering Smoothing (Lowpass) Spatial Filters, Sharpening (High-pass) Spatial Filters, High-pass, Band-reject, and Bandpass Filters from Lowpass Filters, Combining Spatial Enhancement Methods.	
MODULE 3: IMAGE ENHANCEMENT IN THE FREQUENCY DOMAIN	**8 Hrs**
Sampling and the Fourier Transform of Sampled Functions, The Discrete Fourier Transform of One Variable, Some Properties of the 2-D DFT and IDFT, The Basics of Filtering in the Frequency Domain, Image Smoothing Using Low Pass Frequency Domain Filters, Image Sharpening Using High-pass Filters, The Fast Fourier Transform.	
MODULE 4: IMAGE RESTORATION AND MORPHOLOGICAL IMAGE PROCESSING	**7 Hrs**
A Model of the Image Degradation/Restoration process, Noise Models, Restoration in the Presence of Noise Only—Spatial Filtering, Erosion and Dilation, Opening and Closing, The Hit-or-Miss Transform**,** Some Basic Morphological Algorithms.	
MODULE 5: IMAGE SEGMENTATION	**8 Hrs**
Point, Line, and Edge Detection, Thresholding, Segmentation by Region Growing and by Region Splitting and Merging, Region Segmentation Using Clustering and Super pixels, Region Segmentation Using Graph Cuts, Segmentation Using Morphological Watersheds, Case Study: The Use of Motion in Segmentation.	

TEXT BOOK:

1. Rafel C Gonzalez and Richard E. Woods, “Digital Image Processing”, 3rd Edition, Pearson Education, 2010.
2. A. K. Jain, “Fundamentals of Digital Image Processing”, Pearson, 2004.

REFERENCES:

1. Scott.E.Umbaugh, “Computer Vision and Image Processing”, Prentice Hall, 1997.
2. Kenneth R. Castleman, _Digital Image Processing‘, Pearson, 2006.
3. Rafael C. Gonzalez, Richard E. Woods, Steven Eddins, _Digital Image Processing using MATLAB‘, Pearson Education, Inc., 2011.
4. D,E. Dudgeon and RM. Mersereau, _Multidimensional Digital Signal Processing‘, Prentice Hall Professional Technical Reference, 1990.
5. William K. Pratt, _Digital Image Processing‘, John Wiley, New York, 2002
6. Milan Sonka et al _Image processing, analysis and machine vision‘, Brookes/Cole, Vikas Publishing House, 2nd edition, 1999.

SEMESTER	**VI**					
YEAR	**III**					
COURSE CODE	**21CS3610**					
TITLE OF THE COURSE	**HUMAN COMPUTER INTERFACE**					
SCHEME OF INSTRUCTION	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/ Projects Hours	Total Hours	Credits
	3	**-**	**-**	**-**	**39**	**3**

Perquisite Courses (if any)			
#	Sem/Year	Course Code	Title of the Course
*	*	**	***

COURSE OBJECTIVES:

- Learn the foundations of Human Computer Interface
- Be familiar with the design technologies for individuals and persons with disabilities Be aware of mobile HCI
- Learn the guidelines for user interface

COURSE OUTCOMES:

CO No.	Outcomes	Bloom's Taxonomy Level
CO1	Recognize and Analyze the basics of Human Computer Interface, Ergonomics, and style elements with paradigms.	L3
CO2	Outline the knowledge about the navigational design, evaluation techniques, software process life cycle, golden design rules and guidelines.	L2
CO3	Relate the cognitive and collaborative model, Norman's principles and interaction with case studies	L3
CO4	Design a mobile responsive GUI, elements of mobile design tools -and web interfaces with case studies	L3

CO5	Implement the conversational interface and similar tools to apply in real world applications.	L3

COURSE CONTENT:	
MODULE 1: HCI INTRODUCTION	**7 Hrs**
The Human: I/O channels – Memory – Reasoning and problem solving; The computer: Devices – Memory – processing and networks; Historical evolution of HCI; Interaction: Models – frameworks – Ergonomics – styles – elements – interactivity- Paradigms.	
MODULE 2: SOFTWARE PROCESS, MODELS AND THEORIES	**9 Hrs**
HCI in software process – software life cycle – usability engineering – Prototyping in practice – design rationale. Cognitive models –Socio-Organizational issues and stake holder requirements – Communication and collaboration models. Keystroke level model (KLM), GOMS, CASE STUDIES. Shneiderman's eight golden rules; Norman's seven principles; Norman's model of interaction; Neilsen's ten heuristics with example of use.	
MODULE 3: GETTING STARTED WITH GAME DEVELOPMENT	**8 Hrs**
Create Folders- Importing Textures and Meshes- Configuring Meshes - Planning and Configuring Textures**-** Building Sprites - Importing Audio - Create Prefabs - Scene Building Lighting and Lightmapping - Building a Navigation Mesh.	
MODULE 4 : EVENT HANDLING & PLAYER CONTROLLER	**8 Hrs**
Event Handling – Notifications Manager – Send Message and Broadcast Message Character - Controllers and the First Person Controller - Beginning the Universal First Person Controller - Handling Cash Collection - Life and Death: Getting Started**.**	
MODULE 5: CONVERSATIONAL INTERFACE CASE STUDY	**6 Hrs**
Conversational Interfaces, IVR, Chatbot, ALEXIA, MONTANA and similar tools - Case Studies.	

TEXT BOOKS:

1. Alan Dix, Inc, Finlay, Gregory Abowd, Russell Beale, "Human Computer Interaction", 3rd Edition, Pearson Education, 2004
2. Pro Unity Game Development with C#, Alan Thorn, Apress Berkeley, CA Publisher,

ISBN 978-1-4302-6746-1, 2014.
3. Bill Scott and Theresa Neil, “Designing Web Interfaces”, First Edition, O‟Reilly, 2009

REFERENCES:

1. Interaction Design, beyond Human Computer Interaction”, by I Jennifer Preece, Yvonne Rogers, Helen Sharp, John Wiley & Sons.
3. Brian Fling, “Mobile Design and Development”, First Edition, O‟Reilly Media Inc., 2009.

SEMESTER	VI					
YEAR	III					
COURSE CODE	21CS3611					
TITLE OF THE COURSE	UG Research Project-I/Product Development Foundation- I					
SCHEME OF INSTRUCTION	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	-	-	-	6	-	3

Perquisite Courses (if any)			
#	Sem/Year	Course Code	Title of the Course
***	***	***	***

COURSE OBJECTIVES:

- To identify key research questions within a field to carry out research in a team
- To identify and summarize the literature review of the relevant field
- To demonstrate relevant referencing and inculcate new skills in variousaspects of academic writing
- To demonstrate the knowledge and understanding of writing the publication/report
- To showcase the strong evidence on the clarity of the argument,understanding of the selected domain area and presentation of its technical information
- To detail description of the process of carrying out the independent research in written document along with results and conclusions with reference to the existing literature
- To analyze and synthesize the new research findings **COURSE**

OUTCOMES:

CO No.	Outcomes	Bloom's Taxonomy Level
CO1	Develop the research project by selecting an appropriate research problem.	L3
CO2	Compare the papers relevant to the selected problem domain.	L3
CO3	Construct the model and perform the model evaluation and analysis.	L6
CO4	Draft of the Publication or Demonstration of the Proof-of- concept product, Draft of patent application.	L2

COURSE CONTENT:

The research topic proposed by both the guide and the student team should be approved by the department chairman to proceed further. A degree of industrial input and involvement will be encouraged, and can be facilitated through existing academic- industrial collaborations or by addressing specific topics that are of interest to industrial partners.

All projects will be closely supervised by the Project Guide with ongoing feedback and guidance at all stages of the project from the conception to completion.

The following criteria will be checked by the department chairman to approve for the research proposal:

a. Department staff as course guide

1. Ability to provide research direction to the student in the chosen field of interest
2. Ability to design an appropriate research strategy and methodology tocarry out the research by student
3. Ability to provide and evaluate the strong literature review document forthe chosen research topic
4. Ability to train students on research paper / technical writing skills
5. Conduct reviews in regular time period and submit the evaluation to department chairman

b. Student Team

1. To be dedicated and committed to work on a new research topic by learning new technical skills
2. To have fair knowledge on what is product development or research topic
3. To have constant interaction with allocated guide by providing weekly updates
4. To be committed to complete the project and submitting the technical paper within the stipulated time framed by the university

Evaluation:

There will be CIA evaluation as well as the Semester end evaluation of the work done. Itwill be done by a committee of senior researchers of the Department.

SEMESTER	VI					
YEAR	III					
COURSE CODE	21CS3612					
TITLE OF THE COURSE	GAME THEORY					
SCHEME OF INSTRUCTION	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	3	-	-	-	39	3

Perquisite Courses (if any)			
#	Sem/Year	Course Code	Title of the Course
*	**	***	****

COURSE OBJECTIVES:

- To provide a foundation of game theory to help students apply game theory to problem solving in a rigorous way.

COURSE OUTCOMES:

CO No.	Outcomes	Bloom's Taxonomy Level
CO1	Understand the fundamental concepts of non-cooperative and co-operative game theory, in particular standard game models and solution concepts.	L2
CO2	Understand a variety of advanced algorithmic techniques and complexity results for computing game-theoretic solution concepts (equilibria).	L2
CO3	Apply solution concepts, algorithms, and complexity results to unseen games that are variants of known examples	L3
CO4	Understand the state of the art in some areas of algorithmic research, including new developments and open problems.	L2

COURSE CONTENT:	
MODULE 1: INTRODUCTION	8Hrs
Game Theory, Games and Solutions Game Theory and the Theory of Competitive Equilibrium, Rational Behavior, The Steady State and Deductive Interpretations, Bounded Rationality Terminology and Notation Nash Equilibrium- Strategic Games, Nash Equilibrium Examples Existence of a Nash Equilibrium, Strictly Competitive Games, Bayesian Games: Strategic Games with Imperfect Information	
MODULE 2: MIXED, CORRELATED, AND EVOLUTIONARY EQUILIBRIUM	8Hrs

Mixed Strategy Nash Equilibrium Interpretations of Mixed Strategy Nash Equilibrium Correlated Equilibrium Evolutionary Equilibrium Rationalizability and Iterated Elimination of Dominated Actions-Rationalizability Iterated Elimination of Strictly Dominated Actions, Iterated Elimination of Weakly Dominated Actions	
MODULE 3: KNOWLEDGE AND EQUILIBRIUM	**7Hrs**
A Model of Knowledge Common Knowledge, Can People Agree to Disagree, Knowledge and Solution Concepts, The Electronic Mail Game	
MODULE 4: EXTENSIVE GAMES WITH PERFECT INFORMATION	**7Hrs**
Extensive Games with Perfect Information Subgame Perfect Equilibrium Two Extensions of the Definition of a Game the Interpretation of a Strategy, Two Notable Finite Horizon Games Iterated Elimination of Weakly Dominated Strategies Bargaining Games - Bargaining and Game Theory, A Bargaining Game of Alternating Offers Subgame Perfect Equilibrium Variations and Extensions.	
MODULE 5: REPEATED GAMES	**9Hrs**
The Basic Idea Infinitely Repeated Games vs. Finitely Repeated Games Infinitely Repeated Games: Definitions Strategies as Machines Trigger Strategies: Nash Folk Theorems Punishing for a Limited Length of Time: A Perfect Folk Theorem for the Limit of Means Criterion Punishing the Punisher: A Perfect Folk Theorem for the Overtaking Criterion Rewarding Players Who Punish: A Perfect Folk Theorem for the Discounting Criterion The Structure of Subgame Perfect Equilibria Under the Discounting Criterion Finitely Repeated Game.	

TEXT BOOKS:

1. Martin J. Osborne, Ariel Rubinstein. A Course in Game Theory. The MIT Press, August 1994.
2. Y. Narahari. Game Theory and Mechanism Design, IISc Press and the World Scientific Publishing Company, March 2014

REFERENCES:

1. Andrés Perea, Epistemic Game Theory: Reasoning and Choice. Cambridge: Cambridge University, July 2012.
2. Michael Maschler, Eilan Solan, and Schmuel Zamir. Game Theory. Cambridge University Press, 2013.
3. Roger B. Myerson. Game Theory: Analysis of Conflict. Harvard University Press, September 1997

SEMESTER	**VI**					
YEAR	**III**					
COURSE CODE	**21CS3613**					
TITLE OF THE COURSE	**DATA SCIENCE**					
SCHEME OF INSTRUCTION	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	3	**-**	**-**	**-**	**39**	**3**

Perquisite Courses (if any)			
#	Sem/Year	Course Code	Title of the Course
*	**	**	***

COURSE OBJECTIVES:

- To Understand the concept of Data Preprocessing and Transformation
- To use statistical and computational techniques to Discover, Analyze, Visualize andPresent Data
- To analyse the data using visual & summary analytics and common probability distributions
- To acquire the knowledge about building and interpreting regression models and classification with one or more predictors
- To Applying Unsupervised learning approach to the applications

COURSE OUTCOMES:

CO No.	**Outcomes**	**Bloom's Taxonomy Level**
CO1	Apply statistical techniques to preprocess data and perform exploratory data analysis.	L3
CO2	Understand and be able to use appropriate statistical andmachine learning modeling techniques for data analysis and Modeling	L2
CO3	Use data visualization techniques for exploratory dataanalysis and communicating the results	L3
CO4	Evaluate and improve performance of Models	L5
CO5	Use appropriate python libraries for data preprocessing, analysis, modeling and visualization	L3

COURSE CONTENT:	
MODULE 1	**8Hrs**

Overview of the Data Science process. Statistical thinking in the age of big data – Population and Samples, Measures of central tendencies, variability, hypothesistesting, correlation, statistical models and inference. Basic Python constructs and data structures, Jupyter Notebooks	
MODULE 2	**8Hrs**
Data Preprocessing: Data Cleaning - Missing values, Noisy data, Data cleaning process, data Reduction: Principal Components Analysis, Data Transformation: Strategies Overview, Data Transformation by normalization, Discretization by binning. Introduction to Pandas for Data Wrangling.	
MODULE 3	**8Hrs**
Exploratory Data Analysis and Data Visualization with Python: Introduction, Scatter Plots, Histogram, Box Plots, Violin Plot, Heat Map, waffle charts, word clouds, attractive regression plots. Visualizing geospatial data using Folium. choropleth maps. Case Study: Let my dataset change your mindset by Dr HansGosling.	
MODULE 4	**7 Hrs**
Basic Machine Learning Algorithms – Linear Regression, k-nearest neighbors, k- means,decision trees, naïve Bayes	
MODULE 5	**8Hrs**
Model Evaluation: Confusion Matrix, Evaluation Measures. Comparing Classifiers based on cost-benefit and ROC curves. Improving Classifier accuracy: Ensemble Methods, Bagging and Boosting. **Capstone Project**	

TEXT BOOKS:

1. Cathy O'Neil and Rachel Schutt, Doing Data Science, O'reilly Publications,2014
2. Jiawei Han, MichelineKember and Jian Pei, Data Mining Concepts and Techniques, 3rd edition, Elsevier,2012

REFERENCES:

1. Jake VanderPlas, Python Data Science Handbook – Essential tools for working with data, O'Reilly,2016
2. Data Science and Big Data Analytics, Wiley Publications,2015
3. Hastie, Tibshirani, Friedman, "The Elements of Statistical Learning"(2nd edition),Springer,2008

SEMESTER	**VI**					
YEAR	**III**					
COURSE CODE	**21CS3614**					
TITLE OF THE COURSE	**BIG DATA ANALYTICS**					
SCHEME OF INSTRUCTION	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	3	**-**	**-**	**-**	**39**	**3**

Perquisite Courses (if any)			
#	Sem/Year	Course Code	Title of the Course
*	**	**	***

COURSE OBJECTIVES

- **To optimize business decisions and create competitive advantage with Big Data analytics**
- **To explore the fundamental concepts of big data analytics**
- **To learn to analyze the big data using intelligent techniques**
- **To understand the various search methods and visualization techniques.**
- **To learn to use various techniques for mining data stream**
- **To understand the applications using Map Reduce Concepts**
- **To introduce programming tools PIG & HIVE in Hadoop eco system.**

COURSE OUTCOMES:

CO No.	Outcomes	Bloom's Taxonomy Level
CO1	Outline the Hadoop related tools for big data analytics and perform basic Hadoop administration.	L1
CO2	Exemplify the concepts of HDFS and MapReduce framework.	L2
CO3	Illustrate the big data using programming tools like Pig and Hive	L3
CO4	Discriminate the big data for useful business applications and discuss the data for real time applications	L4
CO5	Interpret the various concepts of HBase, Zookeeper, Regression analysis and Visual data analysis techniques for real time applications	L2

COURSE CONTENT:	
MODULE 1: Introduction to big data	**8 Hrs**

Introduction to Big Data Platform – Characteristics of big data-Data in the warehouse and data in Hadoop- Importance of Big data, Challenges of Conventional Systems, Analytic Processes and Tools - Analysis vs reporting, Introduction To Streams Concepts – Stream Data Model and Architecture - Stream Computing	
MODULE 2: Hadoop	**7 Hrs**
History of Hadoop, Hadoop Distributed File System (HDFS) , Introducing and Configuring Hadoop cluster (Local, Pseudo-distributed mode, Fully Distributed mode), Configuring XML files. Comparison between Hadoop1 and Hadoop2	
MODULE 3: Pig	**8 Hrs**
Hadoop Programming Made Easier, Admiring the Pig Architecture, Data processing operators in Pig, Going with the Pig Latin Application Flow, Working through the ABCs of Pig Latin, Evaluating Local and Distributed Modes of Running Pig Scripts, Checking out the Pig Script Interfaces, Scripting with Pig Latin	
MODULE 4: Hive	**8 Hrs**
Applying Structure to Hadoop Data with Hive: Saying Hello to Hive, Seeing How the Hive is Put Together, Getting Started with Apache Hive, Examining the Hive Clients, Working with Hive Data Types, Creating and Managing Databases and Tables, Seeing How the Hive Data Manipulation Language Works, HiveQL – Querying Data in Hive	
MODULE 5: HBase	**8 Hrs**
Fundamentals of HBase and ZooKeeper, Predictive Analytics- Simple linear regression- Multiple linear regression- Visualizations - Visual data analysis techniques, IBM InfoSphere BigInsights and Streams	

TEXT BOOKS:

1. Tom White "Hadoop: The Definitive Guide" Third Edition, O'reilly Media, 2012.

REFERENCE BOOKS

1. Chris Eaton, Dirk DeRoos, Tom Deutsch, George Lapis, Paul Zikopoulos, "Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data", McGrawHill Publishing, 2012.
2. Bill Franks, "Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics", John Wiley& sons, 2012.
3. Pete Warden, "Big Data Glossary", O'Reilly, 2011.

SEMESTER	**VI**				
YEAR	**IV**				
COURSE CODE	**21CS3615**				
TITLE OF THE COURSE	**QUANTUM COMPUTATION**				
SCHEME OF Instruction	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours
	3	**-**	**-**	**-**	**39**

Perquisite Courses (if any)			
#	Sem/Year	Course Code	Title of the Course
***	***	***	***

COURSE OBJECTIVES:

- To apply techniques of linear algebra to quantum mechanics
- To analyze basic quantum circuits
- To explore the techniques of quantum communication
- To study the protocols of quantum cryptography

COURSE OUTCOMES:

CO No.	**Outcomes**	**Bloom's Taxonomy Level**
CO1	Apply techniques of linear algebra to quantum mechanics problems	L3
CO2	Analyze basic quantum circuits	L4
CO3	Explore the techniques of quantum communication	L2
CO4	Study the protocols of quantum cryptography	L2
COURSE CONTENT:		

MODULE 1: LINEAR ALGEBRA REVIEW	9 Hrs
Bases and Linear Independence, Linear Operators and Matrices, Inner Products Eigen Vectors and Eigen Values, Adjoints and Hermitian Operators, Tensor Products, vii. Operator Functions, viii. Commutator and Anti-Commutator	
MODULE 2: QUANTUM MECHANICS	**8 Hrs**
State Space, Evolution, Measurement, Distinguishing Quantum States, Projective Measurements and POVMs	
MODULE 3: QUANTUM GATES AND ALGORITHMS	**7 Hrs**
Universal set of gates, quantum circuits, Solovay-Kitaev theorem, Deutsch-Jozsa algorithm, Shor's factoring, Grover Algorithm and HHL Algorithm	
MODULE 4 : QUANTUM COMMUNICATION	**8 Hrs**
Overview of Quantum Operations, Quantum Noise , Distance Between Quantum States, Accessible Information , Data Compression , Classical Information Over Quantum Channels , Quantum Information Over Quantum Channels , Entanglement as a Physical Resource	
MODULE 5 : QUANTUM CRYPTOGRAPHY	**8 Hrs**
Private Key Cryptography, Privacy Amplification, Quantum Key Distribution, Privacy and Coherent Information, Security of Quantum Key Distribution	

TEXT BOOK:

1. Nielsen, M. A., & Chuang, I. (2002). Quantum computation and quantum information.

REFERENCES:

1. Phillip Kaye, Raymond Laflamme et. al., An introduction to Quantum Computing, Oxford University press, 2007.
2. Chris Bernhardt, Quantum Computing for Everyone, The MIT Press, Cambridge, 2020

SEMESTER	VI					
YEAR	III					
COURSE CODE	21CS3616					
TITLE OF THE COURSE	MOOC					
SCHEME OF Instruction	Lecture Hours	Tutorial Hours	Practical Hours	Seminar/Projects Hours	Total Hours	Credits
	-	--	-	--	39	3

Course Outcomes:

1. Enabling students to obtain certificates to make students employable in the industry or pursue a higher education program.
2. Relevant exposure to tools and technologies are being offered.

Massive Open Online Courses (MOOCs) – Guidelines & Policy

1. Students shall enroll the MOOC courses that is available on the NPTEL/SWAYAM (Swayam.gov.in) platform whenever it notifies (twice in a year).
2. The list of NPTEL / SWAYAM courses related to Computer Science & Engineering thatis in line with the students interest will be announced at the departmental level for enrollment.
    a. That is, the predefined list of courses is provided by the department to thestudents, and only those courses shall be considered and not others.
3. Students shall also enroll in Coursera / Udemy / Udacity / Infosys Spring Board, where DSU can consider the grades / marks provided by these platforms if they are proctored ones. Examinations are to be conducted by DSU if proctored assessments are not conducted by these platforms.
4. The MOOCs courses option shall be considered only for students having a minimum CGPA of **6.75.**
5. The interested student has to enroll as per the guidelines of the NPTEL / SWAYAM or other platforms mentioned in item 3 within enrollment end date.
6. The credits assigned would depend on the number of weeks. The department shallconsider 12 weeks course to map for 03 Credits.
7. A faculty member shall be appointed as SPOC to keep a track of students undertaking courses and collect certificates from students upon completion on theplatforms mentioned above.
8. Student has to pursue and acquire a certificate for a MOOCs course and after successful completion, the student shall submit the certificate to the Department and credits shall be transferred to the grade card accordingly based on the items 1-3 above.
9. The examination fee for obtaining the certificate shall be borne by the student.
10. In case a student fails to complete the MOOC course, then the student shall repeatthe same on the NPTEL/SWAYAM or other platforms mentioned in item 3 or the student may opt for department elective with permission of the department chair.

11. Following is the proposed range for the award of grades towards the creditstransfer.

Range: Consolidated MOOC Score (Assignment+ Proctored exam)	**Proposed Grade Point**	**Grade**
90-100	10	O
80-89	9	A+
70-79	8	A
60-69	7	B+
55-59	6	B
50-54	5	C
40-49	4	P
Less than 40	0	F