COURSE STRUCTURE & SYLLABUS

FOR

B. TECH. PROGRAMME IN
COMPUTER SCIENCE AND ENGINEERING

Semester 3rd to Semester 8th

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
NORTH CAMPUS
UNIVERSITY OF KASHMIR
November-2021
(Applicable for batch 2020 and onwards)

Code	**Nomenclature**	**Code**	**Nomenclature**
ESC	Engineering Science Course	**PROJ**	Project Work
BSC	Basic Science Course	**IT**	Industrial Training/ Internship
PCC	Professional Core Course	**L : T : P**	Lecture: Tutorial: Practical
PEC	Professional Elective Course	**IA**	Internal Assessment: Assignment + Quiz/Viva Voce (10 Marks) + Punctuality (5 Marks)
OEC	Open Elective Course	**MSE**	Mid Semester Evaluation (35 Marks)
SEM	Seminar	**ISE**	Internal Semester Evaluation: ISE=MSE+IA
IP	Induction Program	**ESE**	End Semester Evaluation (50 Marks)

3rd Semester

S. No.	**Course Type**	**Course Code**	**Course Name**	**Hours**			**Credits**	**Marks**		
				L	**T**	**P**		**ISE**	**ESE**	**Total**
1	PCC	MTH30120	Discrete Mathematical Structures	3	-	-	3	50	50	100
2	PCC	CSE30220	Object Oriented Programming using Java	3	-	-	3	50	50	100
3	PCC	CSE30320	Database Management Systems	3	1	-	4	50	50	100
4	ESC	ECE30420	Digital Systems Design	3	1	-	4	50	50	100
5	ESC	ECE30520	Signals and Systems	3	-	-	3	50	50	100
6	PCC	CSE30220L	Object Oriented Programming Lab	-	-	4	2	50	50	100
7	PCC	CSE30320L	Database Management Systems Lab	-	-	4	2	50	50	100
8	ESC	ECE30420L	Digital Systems Design Lab	-	-	2	1	50	50	100
			Total:	**27**			**22**	**400**	**400**	**800**

4th Semester

S. No.	**Course Type**	**Course Code**	**Course Name**	**Hours**			**Credits**	**Marks**		
				L	**T**	**P**		**ISE**	**ESE**	**Total**
1	PCC	MTH40120	Probability, Statistics and Queuing	3	-	-	3	50	50	100
2	PCC	CSE40220	Web Programming	3	-	-	3	50	50	100
3	PCC	CSE40320	Computer Architecture and Organization	3	1	-	4	50	50	100
4	PCC	CSE40420	Data Structures	3	1	-	4	50	50	100
5	ESC	ECE40520	Communication Systems	3	-	-	3	50	50	100
6	PCC	CSE40220L	Web Programming Lab	-	-	4	2	50	50	100
7	PCC	CSE40420L	Data Structures Lab	-	-	4	2	50	50	100
8	ESC	ECE40520L	Communication Systems Lab	-	-	2	1	50	50	100
			Total:	**27**			**22**	**400**	**400**	**800**

5th Semester

S. No.	Course Type	Course Code	Course Name	Hours			Credits	Marks		
				L	T	P		ISE	ESE	Total
1	PCC	CSE50120	Algorithm Analysis and Design	3	1	-	4	50	50	100
2	PCC	CSE50220	Operating System	3	1	-	4	50	50	100
3	PCC	CSE50320	Software Engineering	3	1	-	4	50	50	100
4	PCC	CSE50420	Python Programming	2	-	-	2	50	50	100
5	PCC	CSE50520	Microprocessor	3	-	2	4	50	50	100
6	PCC	CSE50120L	Algorithm Analysis and Design Lab	-	-	4	2	50	50	100
7	PCC	CSE50220L	Operating System Lab	-	-	2	1	50	50	100
8	PCC	CSE50420L	Python Programming Lab	-	-	4	2	50	50	100
			Total:	**29**			**23**	**400**	**400**	**800**

6th Semester

S. No.	Course Type	Course Code	Course Name	Hours			Credits	Marks		
				L	T	P		ISE	ESE	Total
1	PCC	CSE60120	Theory of Computation	3	1	-	4	50	50	100
2	PCC	CSE60220	Artificial Intelligence	3	-	-	3	50	50	100
3	PCC	CSE60320	Computer Networks	3	1	-	4	50	50	100
4	PCC	CSE60420	Computer Graphics	3	1	-	4	50	50	100
5	PEC	PEC6XX20	Elective-I	3	-	2	4	50	50	100
6	PCC	CSE60220L	Artificial Intelligence Lab	-	-	2	1	50	50	100
7	PCC	CSE60320L	Computer Networks Lab	-	-	2	1	50	50	100
8	PCC	CSE60420L	Computer Graphics Lab	-	-	2	1	50	50	100
9	SEM	CSE60620	Seminar	-	-	2	1	50	-	50
			Total:	**28**			**23**	**450**	**400**	**850**

7th Semester

S. No.	Course Type	Course Code	Course Name	Hours			Credits	Marks		
				L	T	P		ISE	ESE	Total
1	PCC	CSE70120	Cryptography and Network Security	3	1	-	4	50	50	100
2	PCC	CSE70220	Compiler Design	3	-	-	3	50	50	100
3	PEC	PEC7XX20	Elective-II	3	-	2	4	50	50	100
4	PEC	PEC7XX20	Elective-III	3	-	2	4	50	50	100
5	OEC	OEC7X20	Open Elective-I	3	-	-	3	50	50	100
6	PROJ	CSE70320	Pre-Project work	-	1	4	3	50	100	150
7	PCC	CSE70120L	Cryptography and Network Security Lab	-	-	2	1	50	50	100
8	PCC	CSE70220L	Compiler Design Lab	-	-	2	1	50	50	100
			Total:	**29**			**23**	**400**	**450**	**850**

8th Semester

S. No.	Course Type	Course Code	Course Name	Hours			Credits	Marks		
				L	T	P		ISE	ESE	Total
1	PEC	PEC8XX20	Elective-IV	3	-	2	4	50	50	100
2	PEC	PEC8XX20	Elective-V	3	-	2	4	50	50	100
3	OEC	OEC8X20	Open Elective-II	3	-	-	3	50	50	100
4	PROJ	CSE80120	Major Project work	-	1	10	6	100	200	300
5	IT	CSE80220	Industrial Training and Professional Viva	-	-	2	1	50	50	100
			Total:	**26**			**18**	**300**	**400**	**700**

LIST OF ELECTIVES (Elective I, II, III, IV, V and Open electives)

Elective-I (PEC6XX20)	
PEC6**11**20	Advanced Algorithm Design
PEC6**12**20	Software Testing
PEC6**13**20	Numerical Methods
PEC6**14**20	Embedded Systems
PEC6**15**20	Data Mining and warehousing
PEC6**16**20	Enterprise Java

Elective-II and Elective-III (PEC7XX20)	
PEC7**11**20	Machine Learning
PEC7**12**20	Software Project Management
PEC7**13**20	Digital Image Processing
PEC7**14**20	Internet of Things (IoT)
PEC7**15**20	Introduction to Big Data
PEC7**16**20	Advanced Computer Networks
PEC7**17**20	Android Programming
PEC7**18**20	Full stack development-I
PEC7**19**20	Introduction to Data Science
PEC7**20**20	Digital signal Processing

Elective-IV and Elective-V (PEC8XX20)	
PEC8**11**20	Introduction to Deep Learning
PEC8**12**20	Introduction to Robotics
PEC8**13**20	Cloud Computing
PEC8**14**20	Full Stack Development-II
PEC8**15**20	Pattern Recognition
PEC8**16**20	Advanced Computer Architecture
PEC8**17**20	Natural Language Processing
PEC8**18**20	Wireless and Mobile communication
PEC8**19**20	Data Analytics with R
PEC8**20**20	Graph theory

Open Electives (OEC7X20, OEC8X20)	
OEC7120	E-commerce and Cyber Laws.
OEC7220	Latex.
OEC8120	Entrepreneurship and Professional Development.
OEC8220	Multimedia Communications.

Total credits (1st to 8th semester) = 175

THIRD SEMESTER

Course Code: MTH30120	**Course Title:** Discrete Mathematical Structures				
Scheme and Credits-	**L**	**T**	**P**	**Credits**	**Total marks:** 100
Semester: 3^{rd}	3	0	0	3	

Course Objectives: The main objectives of the course are:

- Usage of mathematically correct terminology and notation.
- Introduce concepts of mathematical logic for analysing propositions and proving theorems.
- Use sets for solving applied problems and use the properties of set operations algebraically.
- Introduce basic concepts of graphs, digraphs and trees.
- Model problems in Computer Science using trees and graphs.

UNIT-I No. of Lectures: 12

Relations: Sets, product sets, relations, representation of relations, composition of relations, partitions, equivalence relations.

Counting & Combinatorics: Counting, Sum and product rule, Principle of Inclusion Exclusion. Pigeon Hole Principle, Counting by Bijections. Double Counting. Linear Recurrence relations - methods of solutions. Generating Functions. Permutations and counting.

Ordered Sets and Lattices: Ordered sets, diagram of partially ordered sets, Supremum and Infimum, well-ordered sets, lattices, bounded and complemented lattice, and distributive lattice.

UNIT-II No. of Lectures: 10

Propositional Calculus: Statements, basic operations, truth value of compound statements, algebra of propositions, tautologies and contradiction, conditional and bi-conditional statements, logical implications, logical equivalence, predicates, universal and existential quantifiers. Propositional logic for knowledge representation.

UNIT-III No. of Lectures: 12

Graph Theory: Graphs and multi-graphs, degree of a vertex, paths connectivity, cut points bridges, walks, paths, cycles, connected graphs, bipartite, regular, planar and connected graphs, components, Euler graphs, Euler's theorem, Hamiltonian path and circuits, graph coloring, chromatic number, isomorphism and homomorphism of graphs, Konigsberg seven bridge problem, shortest path. Applications of Graph theory in networking.

Trees: Properties of trees, pendant vertices in trees, degree sequences in trees, necessary and sufficient conditions for a sequence to be a degree sequence of a tree.

UNIT-IV No. of Lectures: 12

Group Theory: Groups, semi group, infinite group, finite group, order of a group, Abelian group, subgroup, necessary and sufficient condition for a subset to be a subgroup of a group, Lagrange's theorem, cosets, normal subgroups, order of an element of a group, cyclic group. Rings, homomorphism and isomorphism of rings.

List of Books

- Elements of Discrete Mathematics by C. L. Liu, 2nd Ed.Tata Mc-Graw Hill.
- Discrete Mathematical Structures, Kolman, Busby and Ross, 6th Ed.PHI (2009).
- Narsingh Deo: Graph Theory with Applications to Engineering and Computer Sciences,PHI.
- Murry R. Spiegel: Discrete Mathematics (Schaums Outline series) Tata McGraw Hill (2009).
- Kenneth H. Rosen: Discrete Mathematics and its applications,5th Ed. Tata McGrawHill.
- K.R Parthasarathy : basic Graph Theory, Tata Mc-Graw Hill.

Course Code: CSE30220	**Course Title:** Object Oriented Programming using Java				
Scheme and Credits-	**L**	**T**	**P**	**Credits**	**Total marks:** 100
Semester: 3rd	3	0	0	3	

Course Objectives:

This course introduces fundamentals of object-oriented programming in Java including defining classes, invoking methods, and using class libraries. Students will be able to design and develop java programs with object-oriented paradigm including abstraction, encapsulation, inheritance and polymorphism.

UNIT-I No. of Lectures: 10

Introduction to object-oriented programming, Basics of java programming, features of java programming, Data types, Variables, Operators, Control statements, Arrays in java, Basics of objects and classes in java, Visibility modifiers, Methods, Constructors, Method overloading, Static members, Use of this reference.

UNIT-II No. of Lectures: 12

Inheritance in java, Overriding, Use of super, Abstract class, Interfaces, Packages, Importing packages, Exception handling in java, Use of try, catch, finally, throw in exception handling. Multithreaded programming, Thread life cycle, Thread priorities, Thread synchronization.

UNIT-III No. of Lectures: 12

Event handling in java, Event types, Mouse and key events, GUI Basics, Panels, Frames, Layout Managers: Flow Layout, Border Layout, Grid Layout, GUI components like Buttons, Check Boxes, Radio Buttons, Labels, Text Fields, Text Areas, Combo Boxes, Lists, Scroll Bars, Sliders, Windows, Menus, Dialog Box, Introduction to swing, Swing vs AWT, Hierarchy of swing components, Containers, JFrame, JApplet, JDialog, JPanel.

UNIT-IV No. of Lectures: 12

Java Files and I/O, Streams, I/O classes, Scanner class, Text and Binary I/O.

Introduction to java collections, Collection classes, Array list, Tree list, Hash table, Enumeration, String tokenizer.

JDBC introduction, JDBC drivers, connecting to database, querying a database.

List of Books

1. Head First Java, O'rielly publications.
2. Introduction to Java Programming (Comprehensive Version), Daniel Liang, Pearson.
3. Java: The complete reference, 12th edition, Herbert Schildt. McGraw Hill.
4. Thinking in Java by Bruce Eikel.

Course Code: CSE30320	**Course Title:** Database Management Systems				
Scheme and Credits:	**L**	**T**	**P**	**Credits**	**Total marks:** 100
Semester: 3rd	3	1	-	4	

Course Objectives: The course objectives are:

- To understand the different issues involved in the design and implementation of a database system.
- To study the physical and logical database designs, database modelling and relational model.
- To understand and use data manipulation language to query, update, and manage a database.
- To develop an understanding of essential DBMS concepts such as: database security, integrity, concurrency, recovery
- To design and build a simple database system and demonstrate competence with the fundamental tasks involved with modelling, designing, and implementing a DBMS.

UNIT-I No. of Lectures: 14

Introduction and Architecture: The Role of Information Systems in Organizations, Overview of database management system, database system vs file system, database characteristics, database system concept and architecture, data independence.

Data Modeling using the Entity Relationship (ER) Model: ER diagram design, Mapping constraints, keys, relationships. generalization, aggregation, Relational Database design Using ER-to-Relational Mapping.

Relational Data Model and Language: Relational data model concepts, integrity constraints.

R**elational algebra:** Operations and Queries.

UNIT-II No. of Lectures: 15

SQL-a relational database language: DML, DDL and DCL. Schema definition in SQL, constraints, views, indexes, queries and sub-queries in SQL, triggers, cursors and stored procedures.

Schema refinement: Normalization, functional dependency, reasoning about functional dependencies, Inference Rules, Equivalence, Minimal Cover, Finding candidate keys. Normal forms based on primary keys (1NF, 2NF, 3NF & BCNF), lossless join and dependency preserving decomposition, multivalued functional dependencies.

UNIT-III No. of Lectures: 14

Transaction Management: Transaction concept and state, implementation of atomicity, isolation and durability, concurrent executions, serializability of schedules.

Concurrency and Recovery Control: Techniques, Lock based protocols, Timestamp based Protocols, Deadlock and deadlock handling.

Recovery: Recovery system, failure classification, storage structure, recovery and atomicity, log based recovery.

UNIT-IV No. of Lectures: 13

Physical storage structures: Data on External Storage, File Organization and Indexing, Cluster Indexes, Primary and Secondary Indexes, Index data Structures, Hash Based Indexing, Tree base Indexing, Comparison of File Organizations, B+ Trees: A Dynamic Index Structure
Introduction to NoSQL databases.

List of Books:

1. R. El. Masri and S. B. Navathe. Fundamentals of Data Base Systems, Benjamin Cummings.
2. H. F. Korth and A. Silberschatz. Database Concepts, 2nd Edition, Mcgraw Hill, 1991.
3. J. D. Ullman. Principles of Database and Knowledge Base Systems, Vol. I & II.
4. Database Systems: A Practical Approach to design, Implementation and Management". Thomas Connolly, Carolyn Begg; Third Edition, Pearson Education.

Course Code: CSE30420	**Course Title:** Digital Systems Design				
Scheme and Credits-	**L**	**T**	**P**	**Credits**	**Total marks:** 100
Semester: 3rd	3	1	0	4	

Course Objectives: The course objectives are:

- To understand the essential knowledge on the fundamentals of the numbersystem, Boolean Algebra, digital circuits, and design principles of digital computing systems.
- To use VHDL for realising basic sequential and combinational circuits.

UNIT-I No. of Lectures: 12

Number system and Codes: Binary, Octal, Hexadecimal number systems and their inter-conversion, Binary Arithmetic, BCD codes, 8421 code, Excess-3 code, Gray code, error detection and correction- Hamming code.

Logic Gates and Boolean Algebra: Basic theorems and properties of Boolean algebra, Boolean functions, Canonical and Standard forms of Boolean expressions, Digital logic gates.

Boolean Function Simplification: The Karnaugh map (K-map) method, Quine McCluskey tabulation method.

UNIT-II No. of Lectures: 14

Combinational circuits: Combinational circuit design, design of half adder/subtractor, full adder/subtractor.

Design of Encoders and Decoders: Priority encoders, BCD to decimal decoder, BCD to seven segment display decoder, design of Multiplexers (MUX).

Semiconductor Memories: Introduction, memory organization, sequential memories, content addressable memories, design of memory modules.

UNIT-III No. of Lectures: 14

Sequential circuit design: Flip Flop and its types, Flip-Flop characteristic and excitation tables, Conversion of flip flops.

Counters and Registers: Design of shift registers, universal shift register, and design of asynchronous counters: Ripple counters, Mod (N) counters, Design of various Synchronous counters: state diagrams and truth tables.

UNIT-IV No. of Lectures: 16

VHDL: Introduction to VHDL terms, Object types, Data types, Operators. Design units in VHDL, Sub programs and Packages, Libraries, IEEE Standard logic, Concurrent Statements, Sequential Statements. Compilation and Simulation of VHDL Code System Design using VHDL. Realization of simple combinational and sequential circuits using VHDL (Adder, Subtractor, Multiplexer, Counters)

List of Books:

1. Malvino and Leach “Digital principles and Applications" Tata McGraw Hill.
2. Vahid, Frank. "Digital Design", any edition, John Wiley and Sons.
3. Mano M Morris, “Digital Design” Pearson Education.
4. James W. Bignell and Robert Donovan, “Digital Electronics"
5. Tokheim, Roger L. Digital Electronics: Principles and Applications, Student Text with MultiSIMCD-ROM. McGraw-Hill, Inc., 2007.
6. J. Bhasker, “A VHDL Primer”, Pearson.

Course Code: ECE30520	**Course Title:** Signals and Systems				
Scheme and Credits-	**L**	**T**	**P**	**Credits**	**Total marks:** 100
Semester: 3rd	3	-	-	3	

Course Objective: The course objectives are:

- Understand the various classification of both continuous and discrete time signals and system.
- Analysis of signals using Laplace Transform and Fourier Transform, analysis and characterization of the discrete signals through Fourier.
- Use probabilistic characterizations of random signals and noise, and measurement derived from these signals.

UNIT I No. of Lectures: 12

Introduction to Signals: Classification of signals-deterministic, non-deterministic, periodic, aperiodic, even, odd signals, energy, power elementary, exponential, sinusoidal, impulse, step, ramp, pulse, square wave signals.

Basic operations on Signals: Time shifting. time scaling, time inversions of signals.

Basic properties of System: causality, time invariance, Stability, linearity, memory, order of system, interconnection of systems.

LTI Systems: Linear time invariant systems, characterization, unit impulse response, convolution, properties of LTI systems.

UNIT II No. of Lectures: 12

Fourier analysis of signals and system: Fourier series of periodic signals and its properties, Fourier transform of aperiodic signals and its properties, Fourier transform of periodic signals, convolution in time and frequency domain , energy and signals, Parseval’s theorem energy spectral density and its properties, Transfer function of LTI system.

Laplace Transform: Relation between Laplace and Fourier transforms, region of convergence, properties of Laplace transform. initial and final value theoren1s. convolution, transfer function of LTI system, concept of poles and zeroes, stability criteria.

UNIT III No. of Lectures: 12

Random Variable Theory and Random Signals: Probability, conditional probability, statistical independence, random variables, discrete and continuous random variables, probability distribution and probability density functions, statistical averages of random variables, some important density functions.

UNIT IV No. of Lectures: 10

Random processes and Characterization: Ensemble and time averages, stationary and non-stationary random process, wide sense stationary random process, autocorrelation and cross correlation functions. Response of LTI systems to random inputs.

Noise: Noise and its types, white noise, signal to noise ratio of LTI systems.

List of Books:

1. Signals, Systems & Communications - B.P. Lathi, BSP.
2. Signals and Systems - A.V. Oppenheim, A.S. Willsky and S.H. Nawabi.
3. Signals and Systems – Simon Haykin and Van Veen, Wiley.
4. Signals and Systems – A. Rama Krishna Rao, 2008, TMH.
5. Signals, Systems and Transforms - C. L. Philips, J.M.Parr and Eve A.Riskin, PE.
6. Signals and Systems – K. Deergha Rao, Birkhauser, 2018.

Course Code: CSE30220L	**Course Title:** Object Oriented Programming using Java Lab				
Scheme and Credits-	**L**	**T**	**P**	**Credits**	**Total marks:** 100
Semester: 3rd	-	-	4	2	

Lab outcomes: After the completion of this course students will be able to:

- Implement object-oriented concepts with Java, including defining classes, invoking methods, using class libraries, etc.
- Develop java programs using the concepts of inheritance, abstract class, interfaces and packages.
- Apply the concepts of multithreading and exception handling to develop efficient codes.
- Design event driven GUI applications.

List of experiments

1. Write a C++ program to declare a class, initialize and display the contents of the class member.

2. Use Eclipse or Net bean platform, familiarization with the various menus. Write a program in java to find the sum of two numbers. Create and initializes an int array, calculate and display the average of its values in average method.

3. Write a Program in java to implement inheritance. Create a class called Employee whose objects are records for an employee. This class will be a derived class of the class Person.

4. Write a Java program to create an abstract class named Shape that contains two integers andan empty method named print Area (). Provide three classes named Rectangle, Triangle, and Circle such that each one of the classes extends the class Shape. Each one of the classes contains only the method print Area () that prints the area of the given shape.

5. Write a program to create interface A. In this interface we have two method meth1 and meth2. Implements this interface in another class named MyClass. Create a package named pl and implement this package in c1 class.

6. Write a Java program that implements a multi-thread application that has three threads. First thread generates random integer every 1 second and if the value is even, second thread computes the square of the number and prints. If the value is odd, the third thread will print the value of cube of the number.

7. Write a Java program that creates a user interface to perform various mathematical operations. Use exception handling for handling various errors.

8. Create a test project, add a test class and run it. See how you can use auto suggestions, auto fill. Try code formatter and code refactoring like renaming variables, methods, and classes. Try debug step by step with a small program which contains at least one if else condition and a forloop.

9. Write a Java program that works as a simple calculator. Use a grid layout to arrange buttons for the digits and for the +, -,*, % operations. Add a text field to display the result. Handle any possible exceptions like divided by zero.

10. Write a Java program that simulates a traffic light. The program lets the user select one of three lights: red, yellow, or green with radio buttons. On selecting a button, an appropriate message with “Stop” or “Ready” or “Go” should appear above the buttons in selected color.

11. Write a program to get the input from the user and store it into file.

12. Write a Java program that loads names and phone numbers from a text file where the data is organized as one line per record and each field in a record are separated by a tab (\t). It takes a name or phone number as input and prints the corresponding other value from the hash table.

13. Create a class to connect to the MySQL database and perform queries, inserts and deletes. It also prints the metadata (table name, column names) of a query result.

14. Mini Project/Lab Assignment

Note: This is only a suggested list of lab exercise. Instructor may frame additional exercise relevant to the course contents.

Course Code: CSE30320L	**Course Title:** Database Management Systems Lab				
Scheme and Credits-	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 3rd	-	-	4	2	
Lab outcomes: • Exhibit an understanding of designing the database structure and control access to the data. • Transform an information model into a relational database schema and to use a data definition language and/or utility to implement the schema using a DBMS. • Create, secure, maintain, and query a database and enforce integrity constraints on a database. • Exhibit SQL and PL/SQL for performing operations. • Familiarisation with No Sql databases.					

List of experiments

1. Familiarization with various DBMS: MySQL. Microsoft Access. Oracle. SQLite.

2. Data Definition Language: Data types, Creating Tables, Alter, Truncate, Rename, Drop. Performing operations using SQL as well as GUI Interface.

3. Implementation of Constraints like Default Constraint, Primary Key, Foreign Key, Unique Key, Not Null, Check. Implementing using SQL as well as GUI.

4. Data manipulation language: Retrieval of Rows using Select Command, Conditional retrieval of Records, Insert Statement, Update Statement, Delete Statement, Working with Null Values, Matching a Pattern from a Table, Ordering the Result of a Query, Aggregate Functions, Grouping the Result of the Query.

5. Working with Joins: Cartesian product, Equi-Join, Inner Join and Outer Join, Self-join, Nested Queries, Views.

6. Data Control Language: Database Security and Privileges through Grant and Revoke Commands. Commit and Rollback Commands.

7. Implementation of Stored Procedures: Writing PL/SQL code, Stored Procedures.

8. Implementation of Triggers and cursors.

9. Demonstrate the working of No SQL: CRUD operations using MongoDB

10. Suitable Mini Project/Lab Assignment.

Note: This is only a suggested list of lab exercise. Instructor may frame additional exercise relevant to the course contents.

Course Code: ECE30420L	**Course Title:** Digital Systems Design Lab.				
Scheme and Credits-	**L**	**T**	**P**	**Credits**	**Total marks:** 100
Semester: 3rd	-	-	2	1	

Lab outcomes: After the completion of this course students will be able:

- To understand the usage of simulation environments like Multisim, Logisim, etc. for defining andsimulating circuits.
- To Design and analyse combinational and sequential circuits using various trainer kits and IC's.
- To design, develop and model some basic combinational and sequential logic circuits using VHDL/Verilog and implementation of digital circuits on Xilinx.

List of experiments

1. Familiarization with various basic lab apparatus: DSO, Frequency Generators, Multimeters, Breadboard, Trainer kits, etc.
2. Installation of simulation/design tools Xilinx, Multisim etc.
3. Study and verify the truth table of various logic gates, identify various ICs and their specification.
4. Design of various combinational circuits like adders, encoders. Subtractors using gate IC's
5. Verification of truth table of various circuits using corresponding IC's for Multiplexers, De-multiplexer, Flip Flops etc.
6. Design and simulation of various combinational circuits like adder, multiplexer, encoder, decoder, etc. using simulation softwares.
7. Design and simulation of various code converter circuits like binary to grey, grey to binary, etc.
8. Design and simulation of various synchronous circuits like flip flops, counters, registers, sequence detectors etc.
9. Verify the truth table of decoder driver 7447/7448 and operate a 7-segment display through a counter using a low frequency clock.
10. Design and simulation of various simple combinational and sequential circuits using VHDL on Xilinx environment: (Adder, Counters, Sequence detectors, Registers etc.)
11. Mini Project/Lab Assignment

Note: This is only a suggested list of the lab exercise. The instructor may frame additional exercises relevant to the course contents.

FOURTH SEMESTER

Course Code: MTH40120	**Course Title:** Probability Statistics and Queuing				
Scheme and Credits-	**L**	**T**	**P**	**Credits**	**Total marks:** 100
Semester: 4th	3	-	-	3	

Course Objectives: The course objectives are:

1. Calculate probabilities using Conditional probability, Rule of total probability and Bayes' theorem.
2. Understand the concept of a random variable and the probability distributions.
3. To provide the required mathematical support in real life problems and develop probabilistic models which can be used in several areas of science and engineering.

UNIT-I No. of Lectures: 10

Probability: Random experiment, sample space, events, classical statistical and axiomatic definitions of probability. Statements and proof of theorems on addition and multiplication of probabilities, simple problems. Bayes theorem on conditional probability.

UNIT-II No. of Lectures: 12

Measures of central tendency and measures of variations (dispersions), moments, measures of skewness and kurtosis.

Random variables: Derivation of formula for mean variance and moments of random variables for discrete and continuous cases. probability mass, probability density and cumulative distribution functions, mathematical expectation, moments, probability and moment generating function, median and quantiles, Markov inequality, Chebyshev's inequality, problems.

UNIT-III No. of Lectures: 12

Special Distributions: Binomial, Poisson and normal distributions, beta and gamma distribution.

Sampling Distributions: t-distribution, f-Distribution, Chi-square distribution and their applications. Methods of least squares, fitting a straight line and parabola of degree-p. Regression and correlation. Multiple and partial correlation.

UNIT-IV No. of Lectures: 12

Markov Chain and Reliability: Markov chain, transition probabilities, Limiting distributions, concepts of reliability. Introduction to Queuing Theory: Markovian queuing models, Little's formula, application of queuing theory, M/M/l Model.

List of Books:

1. Miller and Fread's Probability and statistics for engineers-Richard A Johnson, Pearson EducationAsia/ PHI.
2. Fundamentals of Mathematical Statistics- S. C. Gupta and V. K. Kapoor, Sultan and Sons, EasternEconomy Edition.
3. Introduction to Mathematical Statistics-Craig and Hogg, Pearson.
4. Probability and Statistics With Reliability, Queuing With Reliability, Queuing and ComputerScience Application by Kishor S. Trivedi, PHI.

Course Code: CSE40220	**Course Title:** Web Programming				
Scheme and Credits-	**L**	**T**	**P**	**Credits**	**Total marks:** 100
Semester: 4th	3	0	0	3	

Course Objectives: The course objectives are:

- Design and develop the front end of the web applications.
- Design and develop the backend of the web applications.
- Provide well organized, reusable and maintainable code.
- Develop highly versatile and stable web programs based on Code Igniter.
- Deploy and host the developed web application.

UNIT-I No. of Lectures: 10

HTML: HTML Documents, Basic structure of an HTML document, Markup Tags, Heading-Paragraphs, Line Breaks, Introduction to elements of HTML, Working with Text, Working with Lists, Tables and Frames, Working with Hyperlinks, Images and Multimedia, Working with Forms and controls.

CSS: Concept of CSS, Creating Style Sheet, CSS Properties, CSS styling (Background, Text Format, Controlling Fonts), Working with block elements and objects, Working with Lists andTables, CSS Id and Class, CSS Color.

UNIT-II No. of Lectures: 12

Bootstrap: History, Fundamentals of Bootstrap, Bootstrap Grid System, Bootstrap Form and Form Components, Introduction Jquery, Element Selector, Document ready function, Events, Event handling with Html or Bootstrap components

JavaScript: JavaScript Basics, JavaScript Events, JavaScript conditions and loop control structures Alert, Prompt and Confirm statements, JavaScript validation.

UNIT-III No. of Lectures: 12

PHP: Introduction, Control statements, Loops, String Functions in PHP, PHP Email Function, Variables Arrays in PHP with Attributes, Date and Time, Image Uploading, File handling in PHP Functions in PHP, Errors handling in PHP, Object Oriented Programming using PHP, Cookies and Session handling. PHP database connectivity to MySQL.

UNIT-IV No. of Lectures: 12

MVC Programming: Introduction, need and applications, MVC architecture, model, viewand controller,

MVC using code Igniter: PHP vs MVC framework, Code igniter architecture, working, views routing, forms, active record, pagination, session.

Introduction to Web Publishing or Hosting: Creating the WebSite, Saving the site, working on the website, Creating website structure, Themes-Publishing web sites.

List of Books:

1. Kogent Learning Solutions Inc. "*HTML 5 in simple steps"*, Dreamtech Press
2. Murray,Tom/Lynchburg, *"Creating a Web Page and Web Site",* College,2002
3. Griffiths, Adam. *CodeIgniter 1.7 Professional Development*. Packt Pub., 2010.
4. Steven M. Schafer HTML, XHTML, and CSS Bible, 5ed Wiley India
5. John Duckett Beginning HTML, XHTML, CSS, and JavaScript Wiley India
6. Sidik, Betha. "FRAMEWORK CODE IGNITER 3." (2018).

<table>
<tr><td colspan="1"><b>Course Code:</b> CSE40320</td><td colspan="5"><b>Course Title:</b> Computer Architecture and Organization</td></tr>
<tr><td><b>Scheme and Credits-</b></td><td><b>L</b></td><td><b>T</b></td><td><b>P</b></td><td><b>Credits</b></td><td rowspan="2"><b>Total marks:</b> 100</td></tr>
<tr><td><b>Semester:</b> 4th</td><td>3</td><td>1</td><td>-</td><td>4</td></tr>
<tr><td colspan="6"><b>Course Objectives:</b> The course objectives are:
• Discuss the basic concepts and structure of computers.
• Understand concepts of register transfer logic and arithmetic operations.
• Explain different types of addressing modes and memory organization.
• Summarize the Instruction execution stages.</td></tr>
</table>

UNIT-I No. of Lectures: 12

Basic Structure, functional interconnection and instruction cycle:

Basic structure and functions of a digital computer, Von Neuman architecture, hierarchical structure of functional units and their interconnection; System bus and its various implementations.

Structure of CPU. CPU registers and their functions viz. Program Counter, Instruction Register, Memory Address Register, Memory Buffer Register, Program Status Word, etc. Execution of a complete (basic) instruction. Instruction set (ISA), Instruction cycle. CISC and RISC.

UNIT-II No. of Lectures: 14

Instruction format and addressing modes:

Various addressing modes, High level applications of addressing modes. Instruction format and types, Opcode and operands. Micro-operations and Register Transfer Language, CPU organizations-accumulator, general register and stack**,** Implications on zero, one, two and three address instructions.

UNIT-III No. of Lectures: 14

Data Representation:

Integer representation-sign-magnitude and two’s complement, Integer addition, subtraction, multiplication and division, Booth’s algorithm, floating point representation, IEEE 754 standard. Datapath design.

Control Unit Organization:

Hardwired control, micro-programmed control, micro-instructions,address sequencing, control memory, Introduction to pipelining.

UNIT-IV No. of Lectures: 16

Memory Hierarchy and Cache Memory Organization: Semiconductor main memory, RAM and its types-SRAM and DRAM, memory hierarchy, locality of reference, cache memory, organization and characteristics, mapping techniques- direct, associative and set associative, replacement algorithms (brief account only), write policies- write-back and write-through.

I/O Techniques and Interfaces and OS support: I/O modules and their functions, programmed I/O, commands, memory-mapped and isolated I/O, interrupt-driven I/O, DMA, Introduction to standard interfaces viz. PCI, SCSI, and USB.

List of Books:

1. Computer System Architecture, M. Moris Mano, Pearson Education.
2. Computer Organisation and Architecture, William Stallings, Pearson Education.
3. Computer System Architecture, J. P. Hayes, Pearson Education.
4. David A. Patterson and John L. Hennessy, Computer Organization and Design: The Hardware/Software Interface, Elsevier.

<table>
<tr><td><b>Course Code:</b> CSE40420</td><td colspan="5"><b>Course Title:</b> Data Structures</td></tr>
<tr><td><b>Scheme and Credits-</b></td><td><b>L</b></td><td><b>T</b></td><td><b>P</b></td><td><b>Credits</b></td><td rowspan="2"><b>Total marks:</b> 100</td></tr>
<tr><td><b>Semester:</b> 4th</td><td>3</td><td>1</td><td>0</td><td>4</td></tr>
<tr><td colspan="6"><b>Course Objectives:</b> The course objectives are:
• To provide the knowledge of basic data structures and their implementations.
• To understand importance of data structures in context of writing efficient programs.
• To develop skills to apply appropriate data structures in problem solving.</td></tr>
</table>

UNIT-I No. of Lectures: 12

Introduction: Data Structures, pointers, primitive and non-primitive data types, abstract data types, memory management techniques.
Arrays: Representation, implementation, polynomial representation. Limitations
Strings: Representation, String operations, Implementing String.h library functions.
Recursion: Introduction, Recursive implementation of Tower of Hanoi.
Stack: Static and dynamic implementation, expression evaluation using stacks, infix, postfix Expressions.

UNIT-II No. of Lectures: 14

Queues: Static and dynamic implementation, circular queue and its implementation, priority queues. **Linked List:** static and dynamic implementation, singly linked list, doubly linked list, circular, multiple liked lists.
Search and Update Operations on Varieties of Linked Lists, Linked List Implementation of Stacks and Queues

UNIT-III No. of Lectures: 15

Trees: Binary trees, binary search trees, static and dynamic implementation of trees, tree operations, insert, delete and search, AVL trees; Introduction to B trees and B+ trees.
Graphs: Representation of graphs, BFS, DFS, applications of DFS and BFS, Topological sorting.
Minimum Spanning Trees (MST), Prim's and Kruskal's algorithms for MST, Connected Components, Dijkstra's Algorithm for Single Source Shortest Paths. Tree Traversals.

UNIT-IV No. of Lectures: 15

Sorting: Insertion sort, bubble sort, selection sort, quick sort, merge sort, heap sort.
Linear Sorting Algorithms: Counting Sort, Radix Sort, Bucket Sort
Searching: Linear search, Binary Search.
Hashing: Introduction, Hash Functions, Hash Table, Closed hashing (open addressing), Linear Probing, Quadratic Probing, Double Hashing, Open hashing (separate chaining).

List of Books:

1. Fundamentals of Data Structures – Horowitz and Sahani, Galgotia Publication.
2. Data Structures Using C and C++ – Aaron M. Tenenbaum, Yedidyah Langsam and Moshe J. Augenstein, PHI Publications.
3. An Introduction to Data Structures with applications – Jean Paul Trembley and Paul G. Sorenson, McGraw Hill Publications.
4. Data Structures and Program Design in C – R. Kruse etal, Pearson Education.
5. Data Structures – Lipschutz, Schaum's Outline Series, TMH.

Course Code: ECE40520	**Course Title:** Communication Systems				
Scheme and Credits-	**L**	**T**	**P**	**Credits**	**Total marks:** 100
Semester: 4th	3	-	-	3	

Course Objectives: The course objectives are:

- Understand analog and digital communication techniques.
- Understand data and pulse communication techniques.
- Understanding various error detection/correction techniques and source coding.

UNIT I No. of Lectures: 15

Analog communication: Introduction to Communication Systems, Modulation and its types. Transmission media-guided medium (twisted pair cable, coaxial, optical fibre), Unguided medium (radio wave, microwave), Amplitude Modulation (DSB-FC)- square law modulator, envelope detector, Double sideband suppressed carrier (DSB-SC)-balanced modulator, ring modulator synchronous detector, Costas loop receiver, Single side band (SSB)-synchronous detector, vestigial side band modulation (VSB), Angle Modulation-frequency, NBFM, WBFM, generation of FM-direct method and indirect method, phase Modulation, NBFM, WBFM, AM and FM Transmitters, tuned radio frequency receiver,
super-heterodyne receiver, FM receiver.

Noise: Noise in AM and FM, SNR, figure of merit calculation.

UNIT II No. of Lectures: 12

Digital communication: Sampling, Quantization, quantization error, pulse code modulation, differential PCM, ADPCM, Delta Modulation, PAM, PWM, PPM

Amplitude Shift Keying (ASK), Frequency Shift Keying (FSK), Minimum Shift Keying (MSK) , Phase Shift Keying (PSK), BPSK, QPSK, 8 PSK, 16 PSK, Quadrature Amplitude Modulation (QAM), 8 QAM, 16 QAM. Bandwidth Efficiency– Comparison of various Digital Communication System (ASK, FSK, PSK, QAM).

UNIT III No. of Lectures: 10

Line coding: Characteristics of line coding, Unipolar-NRZ, Unipolar-RZ, Polar NRZ-L, Polar NRZ-I, PolarRZ, Bipolar AMI, Bipolar pseudoternary, Manchester encoding, Differential Manchester, 2B1Q, 8B6T. **Multiplexing:** Frequency Division multiplexing, Time division multiplexing.

Multiple Access Techniques: FDMA, TDMA, CDMA.

UNIT IV No. of Lectures: 9

Error analysis and Source coding

Error: Types, error detection (Parity check, checksum, CRC) and correction technique (hamming coding)

Source coding: Information, Entropy, Source encoding theorem, Shannon-fano coding, Huffman coding.

List of Books:

1. B. P. Lathi and Z. Ding, Modern Digital and Analog Communication Systems, 4th Edition, Oxford University Press, 2011.
2. S. Haykin, Communication Systems, 4th Edition, Wiley, 2006.
3. J. M. Wozencraft and I. M. Jacobs, Principles of Communication Engineering, Wiley, 1965.
4. Taub & Schilling: Principles of communication systems- McGraw-Hill Education (India).

Course Code: CSE40220L	**Course Title:** Web Programming Lab				
Scheme and Credits-	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 4th	0	0	4	2	

Lab outcomes: After the completion of this course students will be able to,

- Analyse a web page and identify its elements and attributes. Create web pages using HTML and CSS
- Build dynamic web pages using JavaScript (Client-side programming).
- Build dynamic web pages using PHP (Server-side programming).
- Build webpages using Code Igniter Framework, Deploy and host the web Application.

List of experiments

1. Create a static web page demonstrating use of Text formatting tags, marquee, image tags, Inserting Links, Lists, Tables, Frames, form elements.

2. Create static web pages demonstrating external CSS, internal CSS and inline CSS

3. Create static web pages demonstrating use of Text- pseudo classes, Selectors, Links, Backgrounds, and Lists.

4. Design the front end for the examination/result system of any institute which includes the pages given below: Sign up (complete form asking for the details), Log in page, The page for displaying the Marks ineach subject.

5. Create a static web page demonstrating use of bootstrap functionalities like Grid system, Container and container fluid, Alerts, Collapse, Dropdown, and Modals.

6. Write a JavaScript demonstrating control structures, loops, Alert and prompt and confirm statement, Validation of form data.

7. Design the following modules of the examination/result system using Bootstrap and JavaScript.

- Student selecting the various subjects at the time of submitting his examination form
- Display the prompt if the credentials (static) are wrong
- After submitting the form successfully, the prompt should display, “the form has been successfully submitted”.

8. Installation and configuration of XAMP server, Code ignitor

9. Write a PHP code demonstrating use of comments, variables, echo and print, PHP operators, data types, Branching statements, Loops, Arrays.

10. Write a PHP code demonstrating use of PHP functions, passing information between pages, $_GET, $_POST, $_REQUEST, String functions, include and require, Error handling in PHP

11. Demonstrate use of MySQL datatypes, SQL commands-CREATE, UPDATE, INSERT, DELETE, SELECT, PHP functions for MySQL connectivity and operation- mysql_connect, mysql_select_db, mysql_query, Updation and deletion of data using PHP, displaying data from MySQL in webpage, Displaying data from MySQL in webpage.

12. Design the backend of the web pages you have designed in Sr. No. 4 and 7.

- The data of the student should be stored in the database.
- The database should be designed for each table showing the relations between the tables.
- The validation on the data should be done properly.

13. Use Code igniter to demonstrate the implementation of Model, view and controller, working with database covering CRUD operations, Form validation and routing of views.

14. Use Code igniter to demonstrate pagination of data from database, cookie and session management, File uploading and Error handling, Adding JSS and CSS, Sending Email.

15. Develop the examination/result system modules in code Igniter.

Note: This is only a suggested list of lab exercise. Instructor may frame additional exercise relevant to the course contents.

Course Code: CSE40420L	**Course Title:** Data Structures Lab.				
Scheme and Credits-	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 4th	-	-	4	2	
Lab outcomes: After the completion of this course students will: 1. Be able to design and analyse the time and space efficiency of the data structure. · 2. Be capable to identity the appropriate data structure for given problem · 3. Have practical knowledge on the applications of data structures.					

List of experiments

1. Arrays: Program to implement insertion and deletion operations in arrays

2. Stacks:
- Static and dynamic implementation of push, pop and traversal operations in stacks using both arrays and linked list.
- Implementation of stack to convert infix to postfix expression.
- Evaluating mathematical expressions using stack.

3. Queues: Static and dynamic implementation of enqueue and dequeue operations in queues.

4. Linked List:
- Implementation of searching, insertion and deletion operations in linked list.
- Program to implement searching, insertion and deletion operations in doubly linked list.
- Implementation of singly circular and doubly circular linked list.

5. Recursion:
- Demonstrate concept of recursions with problem of tower of Hanoi.
- Implement recursive sorting techniques- merge sort, quick sort.

6. Trees:
- Program to create and traverse a binary tree recursively.
- Program to implement insertion and deletion operations in BST.
- Program to implement insertion and deletion operations in Heaps and Heap Sort.
- Program to implement insertion and deletion operations in AVL (all rotationsmust be included).

7. Searching:
- Program to implement different searching techniques - linear and binary searchand compare their time complexities.
- Implement Binary search using recursion.

8. Sorting: Implementation of different sorting techniques like bubble, selection, insertion, merge, quick and heap sort.

9. Graph: Implement different representations of graph (Adjacency list and adjacencymatrix).
Graph Traversal: Program to implement DFS/BFS of a graph.

10. Hashing:
- Program to implement linear probing.
- Program to implement quadratic probing.
- Program to implement double hashing.
- Program to implement all the functions of a dictionary using hashing.

Note: This is only a suggested list of lab exercise. Instructor may frame additional exercise relevant to the course contents.

Course Code: ECE40520L	**Course Title:** Communication Systems Lab				
Scheme and Credits-	L	T	P	Credits	**Total marks: 100**
Semester: 4th	0	0	2	1	

Lab outcome: At the end of the course students will be able to:

- Implement AM & FM modulation and demodulation
- Implement PSK, ASK schemes
- Implement pulse code modulation
- Visualize effects of sampling
- Implement source coding

List of experiments

1. Familiarization with various Communication Techniques/Technologies using various Trainer Kits like ST22XX, DCSXX.
2. Familiarization with MATLAB, basic programming- matrix addition, subtraction, multiplication, transpose, graph plotting.
3. Verify different cases of AM modulation index. Plotting different waveforms with different modulation index(μ) when $\mu<1$, $\mu=1$, $\mu>1$.
4. Generation and detection of frequency Modulation (FM) signal using hardware kit and MATLAB simulation software. Observing various waveform and their spectrum analysis behaviour.
5. Generation of Phase Shift Keying, Amplitude Shift Keying and Frequency Shift Keying using MATLAB. Observing various waveform and their spectrum analysis behaviour.
6. To observe the spectrum of a different line coding techniques-unipolar, polar, bipolar with the communication trainer kit and MATLAB.
7. To verify the Encoding process of Delta Modulator and corresponding waveform using hardware trainer kit and MATLAB.
8. Verify the sampling theorem for different modulating frequencies fs< 2fm, fs= 2fm and fs>2fm. Reconstruct the original signal from the sampled signal.
9. To study the effect of Companding and compute the quantization SNR using MATLAB.
10. To study generation of Pulse Code Modulation (PCM) and reconstruction of the original analog signal.
11. Implementation of shannon fano coding using MATLAB.
12. Implementation of Huffman coding algorithm using MATLAB.
13. Mini-Project/Lab Assignment.

Note: This is only a suggested list of lab exercise. Instructor may frame additional exercise relevant tothe course contents.

FIFTH SEMESTER

Course Code: CSE50120	**Course Title:** Algorithms Analysis and Design				
Scheme and Credits-	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 5th	3	1	-	4	

Course Objectives: The course objectives are:

- To apply knowledge of computing and mathematics to algorithm design.
- To analyze a problem and identify the computing requirements appropriate for its solution.
- To design, implement, and evaluate an algorithm to meet desired needs.
- To apply mathematical foundations, algorithmic principles, and computer science theory to the modelling and design of computer-based systems in a way that demonstrates comprehension of the trade-offs involved in design choices.

UNIT-I No. of Lectures: 14

Introduction: Algorithm design paradigm, concept of algorithmic efficiency, run time analysis of Algorithms, asymptotic notations, and recurrence relations, master's theorem.
Greedy Method: Overview of the greedy paradigm, minimum cost spanning tree, Prim's algorithm, Kruskal's algorithm, approximate solution (knapsack problem), and single source shortest path, dijkstra's algorithm.

UNIT-II No. of Lectures: 14

Divide and Conquer: Structure of divide and conquer algorithms, binary search and its analysis, analysis of quick sort.
Dynamic Programming: Overview, Divide and Conquer vs Dynamic programming, 0/1 knapsack, Floyd Warshall's algorithm, matrix multiplication, travelling salesman problem, longest common subsequence, Bellman ford algorithm.

UNIT-III No. of Lectures: 14

Backtracking: Overview, N-queen problem, knapsack problem, graph colouring, Hamiltonian cycles. **Branch and Bound:** LC searching, bounding, FIFO branch and bound, Job scheduling, 0/1 knapsack problem, Travelling salesperson problem.
Comparative analysis of Divide and conquer and dynamic programming, backtracking and dynamic programming.

UNIT-IV No. of Lectures: 14

Application of Graph Traversal Techniques: Topological sorting of DAGs, biconnected components, and strongly connected components in directed graphs.
Computational Complexity: Complexity measures, polynomial time complexity, non-polynomial time complexity, NP hard and NP complete classes, polynomial time reduction, Cook's Theorem.

List of Books:

1. Introduction to Algorithms, 4TH Edition, Thomas H Cormen, Charles E Lieserson, Ronald L Rivest and Clifford Stein, MIT Press/McGraw-Hill.
2. Fundamentals of Algorithms – E. Horowitz et al.
3. Algorithm Design, 1ST Edition, Jon Kleinberg and Éva Tardos, Pearson.
4. Algorithm Design: Foundations, Analysis, and Internet Examples, Second Edition, Michael T Goodrich and Roberto Tamassia, Wiley.

<table>
<tr><td><b>Course Code:</b> CSE50220</td><td colspan="5"><b>Course Title:</b> Operating Systems</td></tr>
<tr><td><b>Scheme and Credits-</b></td><td><b>L</b></td><td><b>T</b></td><td><b>P</b></td><td><b>Credits</b></td><td rowspan="2"><b>Total marks: 100</b></td></tr>
<tr><td><b>Semester:</b> 5th</td><td>3</td><td>1</td><td>-</td><td>4</td></tr>
<tr><td colspan="6"><b>Course Objectives:</b><ul><li>Introduces operating system concepts which include process, scheduling, synchronization deadlocks, memory management and I/O management.</li><li>Provides insight of issues to be considered in the design and development of operating system.</li><li>Introduces Unix commands, system call interface for process management,communication and I/O in Unix.</li></ul></td></tr>
</table>

UNIT-I No. of Lectures: 14

Introduction to operating systems: Operating system functions, evolution of operating systems, batch, interactive, time sharing and real-time system, system protection.

Operating system structure: System components, operating system services**.**

Process management: Process concept, principle of concurrency, producer consumer problem, critical section problem, semaphores, inter-process communication.

UNIT-II No. of Lectures: 14

CPU scheduling: Scheduling concept, scheduling criteria, scheduling algorithms, multiprocessor scheduling.

Deadlock: System model, deadlock characterization, prevention, avoidance, detection and recovery.

UNIT-III No. of Lectures: 14

Memory management: Base machine, Resident monitor, Multiprogramming with fixed partition, Multiprogramming with variable partition, Multiple base register, Paging, Segmentation, Virtual memory concept, Demand paging, Performance, Paged replaced algorithm, Allocation of frames, Thrashing, Cache memory, Organization, Impact on performance

Virtual memory management: Virtual memory concept, Demand paging, Performance, Page replacement algorithms, Allocation of frames, Thrashing.

UNIT-IV No. of Lectures: 14

File system: File Concept, File Organization and Access Mechanism, File Directories and File Sharing.

Disk management: Disk Structure, Disk I/O, Disk scheduling algorithms, RAID Structure.

I/O management: I/O devices and organization of I/O function, I/O Buffering, DISK I/O, Operating System Design Issues.

List of Books:

1. J. Peterson, A. Silberschatz, and P. Galvin. Operating System Concepts, Addison Wesley, Edition.
2. M. J. Bach. Design of the Unix Operating System, Prentice Hall of India,1986.
3. Silberschatz, Galvin, Gagne, "Operating System Concepts", John Wiley and Sons.
4. Modern operating systems by A. S. Tanenbaum, PHI
5. Operating system by Milan Milenkovie: Tata McGraw Hill.

Course Code: CSE50320	**Course Title:** Software Engineering				
Scheme and Credits:	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 5th	3	1	0	4	

Course Objectives: The course objectives are:

- To present software engineering as an engineering discipline comprising fundamental principles, models, approaches and tools.
- To understand and apply various analysis, design, testing and review models/techniques, and appreciate the importance of quality and change management.
- To appreciate the quantitative aspect of software engineering by understanding metrics and their application toward estimation of cost and schedule.

UNIT-I No. of Lectures: 14

'Engineering' Concept and SDLC Models: Program vs software. Software characteristics. Why the word 'engineering'? Evolution of software engineering as a discipline. Software crisis. Dissection of IEEE definition of software engineering. Categories, legacy software. Software product- Types and attributes. Software process-a framework. Process models- Waterfall, evolutionary, prototyping, incremental, spiral, RAD. Formal model and its specific application. Concept of agility, Agile development and various approaches. The trend forward-*DevOps* (introduction).

UNIT-II No. of Lectures: 14

Modelling Requirements and Design: Requirements analysis: Feasibility study. Requirements engineering tasks, SRS. Structured and object-oriented analysis including use-cases, data flow diagrams, sequence diagrams, state diagrams, class diagrams, CRC. Formal specifications. Domain analysis.

Design fundamentals- Abstraction, modularity, functional independence, refactoring. Design patterns. Software architecture -Function oriented and object-oriented design, Cohesion, coupling. Graphical notation and Program Design Language. User interface design. Component-based development, software reuse. Software Reliability: Failure and Faults, Reliability Models.

Illustrative case studies involving groups of students demonstrating customer communication, specification, technical reviews, configuration management, test case design.

UNIT-III No. of Lectures: 14

Quality Assurance and Testing: Verification and validation, testing fundamentals. Testing strategies. Functional testing: Boundary value analysis, Equivalence class testing, Decision table testing, Cause effect graphing, Structural testing: Path testing, Data flow and mutation testing, unit testing, integration and system testing, Black box testing, White- box testing, Debugging, Testing Tools & Standards structural and functional testing, regression testing, debugging. Quality control. Software Quality, Management System ISO 9000, SEI CMM, Personal Software Process, Six Sigma, Software Maintenance.

UNIT-IV No. of Lectures: 14

Metrics and Software Cost Estimation: Software cost estimation, cost metrics, LOC and FP quality management, process improvement, configuration management, software reuse, re-engineering, security engineering, service-oriented software engineering, aspect oriented software engineering. Reverse Engineering, Software Reengineering, Configuration Management, Documentation.

List of Books:

1. Software Engineering, A practitioner's Approach- Roger S. Pressman, Mc Graw Hill International Edition.
2. Software Engineering- Sommerville, Pearson Education.
3. The unified modeling language user guide Grady Booch, James Rambaugh, Ivar Jacobson, Pearson Education.

<table>
<tr><td><b>Course Code:</b> CSE50420</td><td colspan="5"><b>Course Title:</b> Python Programming</td></tr>
<tr><td><b>Scheme and Credits:</b></td><td><b>L</b></td><td><b>T</b></td><td><b>P</b></td><td><b>Credits</b></td><td rowspan="2"><b>Total marks: 100</b></td></tr>
<tr><td><b>Semester:</b> 5th</td><td>2</td><td>-</td><td>-</td><td>2</td></tr>
<tr><td colspan="6"><b>Course Objectives:</b> The course objectives are:
• analyse real life situational problems and think creatively about solutions of them.
• apply a solution clearly and accurately in a program using python.
• analyse and visualize the data using python libraries.</td></tr>
</table>

UNIT-I No. of Lectures: 8

Introduction: Introduction to python, programming languages, programming errors.

Variables, expression and statements: identifiers, variables, assignment statements, expressions, named constant, simultaneous assignment, Boolean types, numeric data types, operators, operator precedence and associativity, augmented assignment operators, type conversion and rounding **Conditionals and iteration:** Conditional expressions, random numbers, minimizing numerical errors,if statement, two way if-else, nested if and multi-way if-elif-else statements, for loop, while loop,nested loops, break and continue.

Functions and recursion: Defining a function, function call, return values, positional and keyword arguments, passing arguments by reference values, scope of variables, default arguments, returning multiple values, recursion, recursion vs iteration, tail recursion, math functions.

UNIT-II No. of Lectures: 8

Lists: List basics, copying lists, passing lists to functions, returning lists from functions, searching and sorting lists, multidimensional list. Python Strings.

Tuples, sets and dictionaries: Introduction to tuples, operations on tuples, introduction to sets, set operations, creating dictionary: operations on dictionary.

Files and exceptions: Introduction, text input and output, pickling, exceptions handling.

UNIT-III No. of Lectures: 10

Numpy arrays: Arrays vs Lists, data types, array creation routines, arrays from existing data, indexing and slicing, array manipulation, broadcasting, binary operators, mathematical functions, statistical functions, sort, search and counting functions.

Handling data with pandas: Introduction to pandas, series, data frame, descriptive statistics, sorting, working with csv files, operations using data frames.

Object oriented programming terminology: creating classes, instance objects, accessing attributes, Class Inheritance, Overriding Methods, Data Hiding, Function Overloading.

UNIT-IV No. of Lectures: 10

Data visualization with matplotlib and seaborn: Line plot, multiple subplots in one figure, histograms, bar charts, pie charts, scatter plots; seaborn- color palette, histogram, kernel density estimates, plotting categorical data, facet grid and pair grid.

Introduction to python libraries: Scikit-learn, Tensorflow, PyTorch, OpenCV.

List of Books:

1. Python Programming: A Modern Approach, Vamsi Kurama, Pearson.
2. Learning Python, Mark Lutz, Orielly.
3. Think Python, Allen Downey, Green Tea Press.
4. Core Python Programming, W.Chun, Pearson.
5. Introduction to Python, Kenneth A. Lambert, Cengage.

<table>
<tr><td><b>Course Code:</b> CSE50520</td><td colspan="5"><b>Course Title:</b> Microprocessor</td></tr>
<tr><td><b>Scheme and Credits:</b></td><td><b>L</b></td><td><b>T</b></td><td><b>P</b></td><td><b>Credits</b></td><td rowspan="2"><b>Total marks: 100</b></td></tr>
<tr><td><b>Semester:</b> 5th</td><td>3</td><td>0</td><td>2</td><td>4</td></tr>
<tr><td colspan="6"><b>Course Objectives:</b> The courses objectives are:
• Understanding the architecture and instruction set of typical 8-bit microprocessor (Intel 8085),
• Understanding Input-output techniques and important programmable support chips used in microprocessor-based systems.</td></tr>
</table>

UNIT-I No. of Lectures: 14

Introduction to a Microcomputer System: Architecture of 8-bit Microprocessor, Intel 8085A microprocessor, Pin description and internal architecture.

Operation and Control of Microprocessor: Timing and control unit, op-code fetch machine cycle, memory read/write machine cycles, I/O read/write machine cycles.

Semiconductor Memories: internal structure and address decoding.

UNIT-II No. of Lectures: 14

Instruction Set: Instruction classification-data transfer, arithmetic, logical, branch, stack and machine control groups of instructions.

Addressing Modes: Introduction, types of addressing modes.

Interrupts: Interrupt structure of 8085A microprocessor, vectored and non-vectored interrupts, hardware and software interrupts.

Assembly Language Programs, Logical and mathematical operations, Operations on stack

UNIT-III No. of Lectures: 14

Interfacing: Interfacing of memory chips and I/O devices. Memory mapped and Isolated I/O.

Input/Output techniques: CPU initiated unconditional and conditional I/O transfer, Device initiated interrupt I/O transfer.

Input/Output Ports: Non-Programmable (Intel 8212), Programmable (Intel 8255-Programmable Peripheral Interface).

UNIT-IV No. of Lectures: 14

Introduction to 8086: Pin description and Addressing Modes.

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

Application of microprocessor and Microcontrollers in data acquisition systems, process control, signal processing, data communication, distributed computing and networking.

List of Books:

1. Gaonkar R. S., “Microprocessor Architecture, Programming and Applications”, 5th Ed., Penram International, 2007.
2. Hall D. V., “Microprocessor and Interfacing-Programming and Hardware”, 2nd Ed., Tata McGraw-Hill Publishing Company Limited, 2008.
3. Stewart J, “Microprocessor Systems- Hardware, Software and Programming”, Prentice Hall International Edition, 1990.
4. Mathur, Sunil, “Microprocessor 8086 - Architecture, Programming and Interfacing”, PHI Learning Pvt. Ltd.
5. Muhammad Ali Mazidi, ARM Assembly Language Programming & Architecture.

Course Code: CSE50120L	**Course Title:** Algorithm Analysis and Design Lab				
Scheme and Credits-	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 5th	0	0	4	2	

Lab outcomes: After the completion of this course students will be able to,

- To analyze the running time of asymptotic algorithm.
- To develop algorithms for sorting, searching, insertion and matching.
- To identify and apply the concept of computational intractability.
- To acquire knowledge in NP Hard and complete problem

List of experiments

1. Perform the priori and posterior analysis of below given algorithms and identify the best, average and worst-case complexities. In addition to it, map the complexity with their execution time (Priori vs Posterior).

i. Linear search and binary search.
ii. Bubble sort and selection sort.
iii. Merge sort and quick sort.

2. Greedy method

i. Implement knapsack problem and analyze its complexity.
ii. Implement Minimum Cost Spanning tree using Kruskal’s Algorithm.
iii. Implement Minimum Cost Spanning tree using Prim’s Algorithm.
iv. Compare the complexities of both algorithms on dense and sparse graphs.

3. With the help of a program compare the time complexity of Fibonacci series using recursion and dynamic programming.

4. Single source shortest path

i. Implement single source shortest path using Dijiktra’s algorithm.
ii. Implement single source shortest path using Bellman Ford algorithm.
iii. Compare the run time complexities of both the algorithms. Identify the ways forimproving the performance of these algorithms.

5. All pair shortest path

i. Implement all pairs shortest path using Floyd Warshall’s Algorithm.
ii. Compare single source shortest path and all pair shortest path and analyse their performance. [Hint: apply APSP on all the source nodes of the algorithm and thenapply SSSP on the same graph].

6. Implement LCS problem using Dynamic programming.

7. Implement Knapsack problem using backtracking.

8. Implement Knapsack problem using Branch and bound, Compare the time complexities of the problem using all the three techniques.

9. Implement Job scheduling algorithm using branch and bound.

10. Implement topological sorting of a DAG.

11. Mini Project/Lab assignment.

Note: This is only a suggested list of lab exercise. Instructor may frame additional exercise relevant to the course contents.

Course Code: CSE50220L	**Course Title:** Operating System Lab				
Scheme and Credits:	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 5th	0	0	2	1	

Lab outcomes: After the completion of this course students will be able to:

- Operate Linux environment and resolve interfacing issues in it.
- Simulate operating system concepts like scheduling, deadlock, etc.
- Simulation of CPU Scheduling Algorithms.
- Simulation of Banker's Algorithm for Deadlock Avoidance.

List of experiments

1. Execution of basic UNIX commands:

- File/Directory Handling commands.
- Explore system variables such as PATH, HOME.
- Modifying File Access Permissions and identify different types users in UNIX.

2. Filters and I/O Redirection:

- Demonstrate all features of awk, sed and grep commands. Compare theresults of these commands.
- Execute commands related to inode, I/O redirection, piping.

3. Shell Programming: shell script exercise based on following:

- Interactive shell script, Positional parameters, Arithmetic and Logical operators, If structure
- While, for, until loop, Meta characters.

4. Inter-process Communication:

- Write a program to demonstrate a one-way pipe between two Processes.
- Write a program to illustrate IPC through pipe and fork system calls –Printing only odd numbers.

5. Simulation of scheduling algorithms:

Write a shell program to implement the following process scheduling algorithms

- First Come First Serve
- Shortest Remaining Job First
- Round Robin

6. Producer-Consumer Problem: Implement the Producer – Consumer problem using semaphores (using UNIX system calls).

7. Banker's Algorithm: Write a program to simulate banker's algorithm for deadlock avoidance.

8. Paging and Segmentation:

Write a program to simulate the following memory management techniques: Paging and Segmentation.

9. Mini Project/Lab Assignment.

Note: This is only a suggested list of lab exercise. Instructor may frame additional exercise relevant to the course contents.

Course Code: CSE50420L	**Course Title:** Python Programming Lab				
Scheme and Credits-	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 5th	-	-	4	2	
Lab outcomes: Learn basics of Python, Understand the usage of various python compilers. Apply OOP concepts in Python, Understand the usage of python important python libraries					

List of experiments

1. Installing and configuring the latest version of python and understanding the various differences between the earlier versions.

2. Program set I:

- Write a python program to find the square root of a number (Newton's method).
- Write a python program of exponentiation (power of a number).
- Write a python program to compute the GCD of two numbers.
- Write a python program first N prime numbers.

3. Program set II:

- Write a python program to compute the factorial of a number using recursion.
- Write a program to understand call by reference and call by value in python.
- Write a program to compute the result of the student. (grading system).
- Write a program to implement linear search and binary search.

4. Program set III.

- To create, slice, change, delete and index elements using list, string and tuple.
- To change, delete, add and remove elements in a Dictionary.
- To create a text file , write some data on it and find the most frequent words in that text file.
- To demonstrate the different inbuilt functions in list, tuple, strings and dictionary.

5. To demonstrate the operations using numpy arrays. array creation routines, arrays fromexisting data, indexing, Slicing, binary operators, mathematical functions, statistical functions, sort, search and counting functions.

6. To demonstrate the various operations using pandas library: Data frames and operations, Operations with CSV files.

7. Data refining, applying descriptive statistics (mean, mode, standard deviation.

8. Hands on Python code executing environments like Anaconda/Jupyter Notebook/Spyder or Google colab.

9. Creating the class and objects and working with constructors, Inheritance, Compile time polymorphism vs run time polymorphism, Function overloading and overriding.

10. To demonstrate the visualization of data using matplotlib: Bar Graph, Histogram, Scatter Plot, Area Plot, Pie Chart, subplot.

11. To demonstrate the visualization of data using seaborn**.** Plotting a Displot, Distplot Without the Histogram, KDE, Facet grid and pair grid.

12. To demonstrate the operations using of ScikitLearn Library.

13. To demonstrate the operations using of TensorFlow.

14. To demonstrate the operations using of PyTorch.

15. To demonstrate the operations using OpenCV.

16. To demonstrate the operations using nltk.

17. Suitable mini project/Lab Assignment.

Note: This is only a suggested list of lab exercise. Instructor may frame additional exercise relevant to the course contents.

SIXTH SEMESTER

Course Code: CSE60120	**Course Title:** Theory of Computation				
Scheme and Credits-	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 6th	3	1	-	4	

Course Objectives: The objective of the course is:

- To provide introduction to some of the central ideas of theoretical computer science from the perspective of formal languages and to introduce the fundamental concepts of formal languages, grammars and automata theory.
- To classify machines by their power to recognize languages.
- To understand the differences between decidability and un-decidability.

UNIT-I No. of Lectures: 14

Introduction to Theory of Computation and Finite Automata: Motivation for studying theory of computation, understanding complexity, computability, abstract machine and computations. Notion of formal languages- mathematical notions, Infinite Sets, Closures, Alphabets, Languages & Representations. Language membership problem as the central problem of the subject. Deterministic finite automata (DFA), Non-deterministic finite automata (NFA), and regular expressions- their equivalence/ conversions. Regular grammar and equivalence with FA. Pumping lemma for regular languages, Closure properties of regular languages, Minimization of DFA.

UNIT-II No. of Lectures: 14

Pushdown Automata (PDA) and Context Free Languages (CFL): Context free Grammar (CFG) and CFL. Derivation and parse trees, Ambiguity in CFL, Normal forms of CFL: Chomsky Normal form (CNF) and Greibegh Normal Form (GNF), Closure properties of CFL. Pumping Lemma for CFL. Deterministic and Non-Deterministic PDA. Notion of Acceptance by PDA, Equivalence of PDA and CFG. Introduction to Context sensitive grammar and Context sensitive language.

UNIT-III No. of Lectures: 14

Turing Machines (TM) and Un-decidability: Historical context, Definition of TM, instantaneous description as a snapshot of TM computation, Deterministic and Non- Deterministic TM and their equivalence, Other variants of Turing machine, Un-restricted grammar and its equivalence with TM. Turing-recognizable (recursively enumerable) and Turing-decidable (recursive) languages.

UNIT-IV No. of Lectures: 14

Decidability and Complexity theory: Church Turing thesis, Reducibility and its use in proving un-decidability, The Diagonalization language, Universal Turing machine, Post correspondence problem, Turing's Halting Problem. Chomsky Hierarchy of formal languages.

The Complexity Class P, Satisfiability, The Complexity Class NP, NP Completeness and Reducibility NP complete problems, Cook's Theorem statement.

List of Books:

1. Introduction to Automata Theory, Languages, and Computation, 3nd Edition, John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman, Pearson Education.
2. Introduction to Languages and The Theory of Computation, John C Martin, TMH.
3. Introduction to the Theory of Computation, Michael Sipser, 3rd edition, Cengage Learning.
4. P. Linz, "Introduction to Formal Languages and Automata", PWS Publishing Company.
5. C. Papadimitrou and C. L. Lewis. Elements of Theory of Computation, Prentice-Hall.

Course Code: CSE60220	**Course Title:** Artificial Intelligence				
Scheme and Credits-	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 6th	3	-	-	3	

Course Objectives: The course objectives are:

- Define basic knowledge representation, problem solving, and learning methods of artificial intelligence.
- Apply analytical concepts for solving logical problems using heuristics approaches.
- Understand approaches to syntax and semantics in NLP.
- Develop AI learned concepts using python libraries.

UNIT-I No. of Lectures: 12

Introduction: Introduction to artificial intelligence, history of artificial intelligence, applications of artificial intelligence, intelligent agents, structure of intelligent agents.

Search Techniques: Un-informed search strategies: Breadth first search (BFS), Depth first search (DFS), depth limited search (DLS), uniform cost search, iterative deepening search, hill climbing, beam search. Heuristic Search: Heuristic functions, A* algorithm. Completeness and Optimality of algorithms.

Game Theory: Adversarial search, games, minimax algorithm, optimal decisions in multiplayer games, alpha-beta pruning, evaluation functions.

UNIT-II No. of Lectures: 12

Knowledge Representation and Reasoning: Propositional logic, first-order logic, inference in first order logic- forward and backward chaining, theorem proving, resolution theorem proving, rules and rule-based reasoning, structured knowledge representation. Probabilistic reasoning, reasoning with uncertain knowledge (Bayesian inference networks), Hidden Markov Model

UNIT-III No. of Lectures: 12

Machine Learning introduction: Supervised and Un-supervised Learning. Classification Techniques: Support Vector Machines (SVM), Decision Trees, K-Means clustering.

Statistical Learning Models: Bayes Classifier, Gaussian Mixture Model, Expectation-maximisation algorithm, Naive Bayes model.

Introduction to Neural network architectures, Activation functions.

UNIT-IV No. of Lectures: 10

Genetic Algorithms-operations, applications, limitations. Fuzzy Logic, Fuzzification, Fuzzy Sets, Operations on Fuzzy Sets, Hedges, Reasoning in Fuzzy Logic, Mamdani Inference.

Introduction to NLP, Phases of NLP, construction of parse tree, tokenizing text data, word stemming, word lemmatization, dividing text data into chunks, bag of words, word2Vec.

List of Books:

1. Artificial Intelligence by Kevin Knight, Elaine Rich, B. Shivashankar Nair, Mc Graw Hill.
2. Artificial Intelligence: A Modern Approach by Stuart Russel, Peter Norvig, Pearson.
3. Artificial Intelligence with Python by Prateek Joshi, Packt Publishing.
4. Neural Networks, Fuzzy Logis and Genetic Algorithms: Synthesis, and Applications, S. Rajasekaran, and G. A. Vijayalakshmi Pai, Prentice Hall of India.
5. An Introduction to Genetic Algorithms, Melanie Mitchell, MIT Press.

Course Code: CSE60320	**Course Title:** Computer Networks				
Scheme and Credits-	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 6th	3	1	-	4	

Course Objectives:

- The objective of this course is to gain insight about networks, topologies and the key networking concepts.
- To understand the layered communication architectures (OSI and TCP/IP).
- To understand the basic communication protocols, design issues and significance of each in different layers of ISO and TCP/IP.

UNIT-I No. of Lectures: 14

Basic Concepts of Networks: Advantages and applications, Data transmission techniques: simplex, half duplex, full duplex, synchronous and asynchronous, Types of networks (LAN, MAN and WAN), Different network topologies like bus, star, ring, hybrid.

Network Protocol Architecture: OSI Reference model, Layers of OSI model: Physical, Data-link, Network, Transport, Session, Presentation and Application layer, Concept of TCP/IP Protocol Suite.

Network Switching Techniques: Circuit, message and packet switched networks.

UNIT-II No. of Lectures: 14

Flow and Error Control: Stop and wait flow control, Sliding window flow control, error control protocols, ARQ techniques, Stop-and-wait ARQ, Goback-N ARQ, Selective repeat ARQ. HDLC Protocol, Point to Point Protocol, Ethernet, Token Ring network.

Medium Access Control Protocols: Random Access: ALOHA, Slotted ALOHA, CSMA, CSMA/CD, CSMA/CA; Controlled Access: Polling, token passing, reservation.

UNIT-III No. of Lectures: 14

Network layer: Logical Addressing-IPv4 and IPv6 addresses, Concept of Sub-netting and Classless Addressing, Network Address Translation.

Routing Algorithms: Routing tables, features of a routing algorithm, classification; concept of optimality principle, sink tree, flooding, fixed routing, random routing, adaptive routing; shortest path algorithm, Dijkstra algorithm, distance vector and link state algorithm.

UNIT-IV No. of Lectures: 14

Transport Layer: Process to Process Delivery, Socket Addresses, User Datagram Protocol (UDP), Transmission Control Protocol (TCP), SCTP.

Congestion Control: Congestion in networks and quality of service.

Application Layer: File Transfer, HTTP, Electronic mail, Remote Logging.

List of Books:

1. William Stallings: Data and Computer Communications, PHI
2. D.P Bertsekas “Data Networks” Prentice Hall
3. Andrew Tanenbaum, “Computer Networks”, Pearson.
4. J. F. Kurose “Computer Networking” Addison-Wesley
5. Douglas E. Comer, “Internetworking with TCP/IP, Volume1”, PHI.
6. Douglas E. Comer, “Client-Server Programming with TCP/IP, Volume3”, PHI.
7. Peterson and Davie, “Computer Networks”, Morgan Kaufman

Course Code: CSE60420	**Course Title:** Computer Graphics				
Scheme and Credits-	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 6th	3	1	-	4	

Course Objectives:

The objective of the course is to help students understand the basics of various computer graphics hardware devices. It offers an in-depth exploration of fundamental concepts in 2D and 3D computer graphics like geometric transformations; scan conversion, clipping, visible surface detection besides giving an idea about curves and surfaces. This course presents an introduction to computer graphics designed to give the student an overview of fundamental principles.

UNIT-I No. of Lectures: 14

Introduction to Computer Graphics: Overview of Computer Graphics, Computer Graphics Application and Software, Description of some graphics devices, Active and Passive Graphics Devices, Display Technologies, Storage Tube Graphics Displays, Calligraphic Refresh Graphics Displays, Raster Refresh (Raster-Scan) Graphics Displays, Cathode Ray Tube Basics, Color CRT Raster Scan Basics, Video Basics, The Video Controller, Random Scan Display Processor, LCD displays.

UNIT-II No. of Lectures: 14

Two-Dimensional Transformations: 2D Transformations, Homogeneous Coordinates and Matrix Representation of 2D Transformations, Translation, Rotation, Reflection, Scaling, Combined Transformation, Transformation of Points, Window-to-Viewport Transformations.

Three-Dimensional Transformations: Introduction, Three-Dimensional Scaling, Three-Dimensional Shearing, Three-Dimensional Rotation, Three-Dimensional Reflection, Three-Dimensional Translation, Multiple Transformations.

UNIT-III No. of Lectures: 12

Scan conversion lines, circles and Ellipses: Scan Conversion; line, circle, eclipse. Mid-point criteria, Problems of Aliasing.

Line Clipping algorithms: Cyrus-Beck, Cohen-Sutherland and Liang Barsky.

Clipping Polygons: Sutherland-Hodgeman polygon clipping.

UNIT-IV No. of Lectures: 16

Visible-Surface Determination: Techniques for efficient Visible-Surface Algorithms, Categories of algorithms, Back face removal, The z-Buffer Algorithm, Scan-line method, Painter's algorithms (depth sorting).

Plane Curves and Surfaces: Curve Representation, Nonparametric Curves, Parametric Curves, Cubic Splines, Bezier Curves, B-spline Curves, Bezier Surfaces.

List of Books:

1. Computer Graphics and Applications, Hearn and Baker, Pearson Education.
2. J. D. Foley, A. Van Dam, S. K. Feiner and J. F. Hughes, Computer Graphics-Principles and Practice, Second Edition in C, Pearson Education, 2003.
3. D. Hearn and M. Pauline Baker, Computer Graphics (C Version), Pearson Education, 2nd Edition, 2004.
4. D. F. Rogers and J. A. Adams, Mathematical Elements for Computer Graphics, 2nd Edition, McGraw-Hill International Edition, 1990.
5. Foley and Van Dam. Fundament also Interactive Computer Graphics, Addison-Wesley.

Course Code: CSE60220L	**Course Title:** Artificial Intelligence Lab				
Scheme and Credits-	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 6th	0	0	2	1	
Lab outcomes: After the completion of this course students will be able to, • To provide skills for designing and analysing AI based algorithms. • To enable students to work on various AI tools. • To provide skills to work towards solution of real-life problems. • To train various machine learning models for classification/regression tasks.					

List of experiments

1. Write a Program to Implement Tic-Tac-Toe game using adversarial searching algorithm.
2. Implementation of Knowledge representation schemes.
3. Write a program to solve 8 queens problem using genetic algorithm.
4. Implementation of TSP using heuristic approach.
5. Implementation of A* Algorithm.
6. Implementation of Hill-climbing to solve 8- Puzzle Problem.
7. Implementation of Expert System with forward chaining/ backward chaining.
8. Pre-processing of datasets for natural language processing.
9. Naive Bayes algorithm for classification task (using standard data sets).
10. Create a decision tree from dataset using ID3 algorithm (using standard data sets).
11. Given a dataset which is linearly separable, classify the data points using SVM.
12. Neural network for classification task (using standard data sets).

Note: This is only a suggested list of lab exercise. Instructor may frame additional exercise relevant to the course contents.

Course Code: CSE60320L	**Course Title:** Computer Networks Lab				
Scheme and Credits-	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 6th	-	-	2	1	
Lab outcomes: • Implement and analyse the data link layer protocols. • Implement and analyse routing and congestion issues in a network. • To be able to work with different networking tools.					

List of experiments

1. Introduction of LANs and Network Wire Crimping.
2. Experimental study of common protocols such as HTTP, FTP, SMTP, using network packetsniffers.
3. Experiments with packet sniffers to study the TCP/IP protocols.
4. Introduction of Network Simulators.
5. Introduction to NS2 (network simulator) - small simulation exercises to study behaviour ofvarious protocols.
6. Experiments with NS2 to study behaviour (especially performance) of various protocols.
7. Setting up a small IP network: Configure interfaces, IP addresses and routing protocols.
8. Small exercises in socket programming in C/C++/Java.

Note: This is only a suggested list of lab exercise. Instructor may frame additional exercise relevant to the course contents.

Course Code: CSE60420L	**Course Title:** Computer Graphics Lab				
Scheme and Credits-	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 6th	-	-	2	1	
Lab outcomes: The students would be able to perform various graphics operations and implement various graphic algorithms using C language					

List of experiments

1. Study of Fundamental Graphics Functions in C Language.
2. Implementation of line generation using slope's method, DDA and Bresenhem's algorithms.
3. Implementation of circle generation using Mid-point method and Bresenhem's algorithm.
4. Implementation of ellipse generation using Mid-point method.
5. Implementation of polygon filling using Flood-fill, Boundary fill and Scan-line algorithms.
6. Implementation of 2D transformation: Translation, Scaling, Rotation, Mirror Reflection and Shearing.
7. Implementation of Line Clipping using Cohen-Sutherland algorithm and Bisection Method.
8. Implementation of Polygon Clipping using Sutherland Hodgeman algorithm.
9. Implementation of 3D geometric transformations: Translation, Scaling and rotation.
10. Implementation of Curve generation using Interpolation methods.
11. Implementation of Curve generation using B-spline and Bezier curves.

Note: This is only a suggested list of lab exercise. Instructor may frame additional exercise relevant to the course contents.

Course Code: CSE60620	**Course Title:** Seminar				
Scheme and Credits-	**L**	**T**	**P**	**Credits**	**Total Marks = 50**
Semester: 6th	-	-	2	1	

Objective:

The seminar is aimed to develop confidence and effective communication skills in students for presenting state-of-the-art computer related technologies. Each student shall identify a topic, get approval of the faculty concerned and present the same before an audience comprising of the faculty members and other students. Students are expected to collect enough literature on their chosen topic, study it thoroughly, prepare a power point presentation and write a report.

Evaluation:

The seminar shall be evaluated by the Departmental Committee based on the quality of the content presented and effectiveness of presentation. The emphasis should be on fluent expression, non-verbal gestures and confident answering of audience queries.

SEVENTH SEMESTER

Course Code: CSE70120	**Course Title:** Cryptography and Network Security				
Scheme and Credits-	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 7th	3	1	-	4	

Course Objectives: The course objectives are:

- To understand the basic network security issues, types of attacks and mechanisms to combat such attacks.
- To understand authentication challenges, requirements and authentication algorithms.
- To comprehend public and private key management issues.
- To understand network layer security protocols, transport layer security protocols and web security protocols and to apply algorithms used for secure transactions in real world applications

UNIT-I No. of Lectures: 15

Basic Concepts: Introduction to security attacks, services and mechanisms, classical encryption techniques substitution ciphers and transposition ciphers, cryptanalysis, steganography, Stream and block ciphers.

Modern Block Ciphers: Block ciphers principles, Shannon's theory of confusion and diffusion, fiestal structure, data encryption standard (DES), block cipher modes of operations, Triple DES.

Number Theory and Finite Fields: Introduction to group, field, finite field of the form GF(p), modular arithmetic, prime and relative prime numbers, extended Euclidean algorithm, Fermat's and Euler's theorem, discrete logarithmic problem.

UNIT-II No. of Lectures: 14

Advanced Encryption Standard (AES): AES encryption, decryption and key expansion.

Public Key Cryptography: Principles of public key cryptosystems, RSA algorithm.

Cryptographic Hash Functions: Application of cryptographic hash functions, requirements, Secure Hash Algorithm (SHA-1).

UNIT-III No. of Lectures: 15

Message Authentication Codes: Authentication requirements, authentication functions, message authentication code, HMAC Algorithm.

Digital Signatures: Digital Signature Properties and Requirements, Elgamal Digital Signature Technique, Digital signature standard (DSS).

Key Management and Distribution: Symmetric key distribution, Diffie-Hellman Key Exchange, Public key distribution, X.509 Certificates, Public key Infrastructure.

UNIT-IV No. of Lectures: 12

IP Security: Architecture, Authentication header, Encapsulating security payloads, combining security associations, key management.

Transport Level Security: Introduction to Secure Socket Layer and TLS, HTTPS.

System Security: Introductory idea of Intrusion, Intrusion detection, Viruses and related threats, Firewalls.

List of Books:

1. Cryptography and Network Security-William Stallings, Pearson Education.
2. Cryptography and Network Security-Behrouz A. Forouzan, Tata McGraw-Hill.
3. Network Security- Charlie Kaufman, Radia Perlman, Mike Speciner.
4. Applied Cryptography, Protocols, Algorithms, and Souce Code in C. Schneier, Wiley

Course Code: CSE70220	**Course Title:** Compiler Design

Scheme and Credits:	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 7th	3	-	-	3	

Course Objectives: The objectives of course are:

- To understand the basic concept of compilation design which include its various phases
- To use flex or similar tools to create a lexical analyser and Yacc/Biscon tools to create a parser
- To implement a various parser such as SLR parser without using any compiler generation tools
- To implement a various intermediate code generation, optimizing techniques without using any compiler generation tools

UNIT-I No. of Lectures: 12

Introduction to Compiler: Compiler Structure: Compilers, Analysis of Source Program, Phases of Compiler, Issues in a Compiler Structure, Major Data Structures in Compiler, Bootstrapping and Porting, Various phases of compiler, analysis-synthesis model of compilation.

Finite Automata and Lexical Analysis: Lexical structure of a language, role of lexical analyzer, tokens, patterns and lexemes: specification and recognition. Design of lexical analyzers, finite state machines, regular expressions and their applications to lexical analysis, optimization of DFA-based pattern matchers, LEX- compiler.

The Syntactic Specification: Context free grammars (CFG), capabilities of CFG.

UNIT-II No. of Lectures: 12

Top Down Parsing: Top down parsing methods, grammar simplification, elimination of left recursion,left factoring, grammar ambiguity. Recursive descent parsing and predictive parsers, LL(1) grammar and LL(1) parsing table and algorithm.

Bottom Up Parsing: Shift-reduce parsing, precedence parsing, LR parsers, the canonical collection of LR(0) items, constructing SLR parsing tables, constructing canonical LR parsing tables, constructing LALR parsing tables, an automatic parser generator(YACC).

UNIT-III No. of Lectures: 12

Syntax-Directed Translation: Translation schemes, inherited and synthesized attributes, dependency graph, evaluation order, intermediate code, postfix notation, parse trees, syntax trees, three address code, quadruples and triples. Bottom-up Evaluation of S-attributed Definitions, L-attributed Definitions Translation of various language statements.

Type Checking: Type systems, type expressions, type conversion

Symbol Tables: Data structure for symbols tables, representing scope information.

UNIT-IV No. of Lectures: 10

Code Optimization and Code Generation: Design issues, addresses in the target code, basic blocks and flow graphs, optimization of basic blocks, code generator. Machine-independent optimizations, loop optimization. DAG representation of basic blocks, value numbers and algebraic laws, global data- flow analysis.

Run-time environment and Symbol: Procedure activation, parameter passing, value return, memory allocation, and scope.

List of Books:

1. Aho, Sethi & Ullman, "Compilers: Principles, Techniques and Tools”, Pearson Education.
2. V Raghvan, “Principles of Compiler Design”, TMH.
3. Charles Fischer and Ricard LeBlanc,” Crafting a Compiler with C”, Pearson Education.
4. K. Muneeswaran, Compiler Design, First Edition, Oxford University Press.
5. J.P. Bennet, “Introduction to Compiler Techniques”, Second Edition, Tata McGraw-Hill.

Course Code: CSE70320	**Course Title:** Pre-Project work				
Scheme and Credits:	**L**	**T**	**P**	**Credits**	**Total marks: 150 marks**
Semester: 7th	-	1	4	3	

Course Objectives:

- To hone the creativity of students enabling them to propose novel and innovative project ideas.
- To develop expertise towards expressing a problem, performing existing system/literature survey and specifying all requirements.

Description:

- In the Pre-project, students are expected to choose a problem and perform its requirement analysis. The same work shall be continued, and its complete implementation carried out in the Major Project during the final (8th) semester.
- In the beginning of the 7th semester, students shall be asked to submit synopses/proposals for their projects. The project shall, in very broad terms, involve problem solving through design and development of some IT solution or any theoretical aspect of computing.
- Students may also be specifically encouraged to come up with any novel idea which addresses some local/regional problem such as
    - Solution or a step towards solution to institutional/departmental issues.
    - Solution/procedure towards efficiency in an existing business process of socio-economic importance like agriculture, art, tourism, healthcare etc.
    - Solution/procedure towards efficient delivery of various public services (e-Governance).
- Students shall submit their project synopses in groups rather than individually. Formation of groups by students is essential for developing attributes of team-work and sharing of responsibilities.
- The final allocation of project guides shall be finalized by the Departmental Committee after receiving and analysing all the synopses.
- Students shall devote time to their project work and consult their project guide/s on a regular basis. The time table shall contain slots exclusively for the same.

Evaluation:

- Out of the 50 marks stipulated for Internal Semester Evaluation (ISE), 25 shall be awarded on the basis of continuous assessment by the respective Supervisor, while the remaining 25 marks shall be evaluated during the project presentation to be held before the Departmental Committee.
- The External Semester Evaluation (ESE) shall be held by an approved external examiner. The External Semester Evaluation (ESE) shall be of 100 marks. The break-up of 100 marks shall be as follows:

    Presentation: 20 marks
    Viva-voce: 50 marks
    Report writing based on State of art, fundamentals of topic and its viability: 30 marks

Course Code: CSE70120L	**Course Title:** Cryptography and Network Security Lab.				
Scheme and Credits-	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 7th	0	0	2	1	
Lab outcomes: The students would be able to implement various encryption techniques. The students will also be able to configure and understand firewall features.					

List of experiments

1. Configure and demonstrate use of Traffic monitoring tool such as Wireshark with security perspective.
2. Implement the following encryption and decryption techniques using high level programming languages
   a. Ceaser cipher b. Substitution cipher c. Hill Cipher
3. Implementation of modern block ciphers like DES, AES.
4. Implementation and study of vulnerabilities in RSA algorithm.
5. Implement the Diffie-Hellman Key Exchange mechanism using HTML and JavaScript.
6. Calculate the message digest of a text using the SHA-1 algorithm in JAVA.
7. Calculate the message digest of a text using the MD5 algorithm in JAVA.
8. Study of the features of firewall in providing network security and configuration of Firewall Security in various platforms (Windows/Linux).
9. Implement web security with Open SSL tool kit.
10. Study of different types of vulnerabilities in websites/ Web Applications/Email.

Note: This is only a suggested list of lab exercise. Instructor may frame additional exercise relevant to the course contents.

Course Code: CSE70220L	**Course Title:** Compiler Design Lab				
Scheme and Credits-	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 7th	0	0	2	1	
Lab outcomes: After the completion of this course students will be able to, • To design various components of compiler like Lexical analyser etc. • To Implement various parsing algorithms.					

List of experiments

1. Design a lexical analyser for given language and the lexical analyser should ignore redundant spaces, tabs and new lines. It should also ignore comments. Although the syntax specification states that identifiers can be arbitrarily long, you may restrict the length to some reasonable value. Simulate the same in C language
2. Implement the lexical analyser using JLex, flex or other lexical analyser generating tools
3. Simulate First and Follow of a Grammar
4. Develop an operator precedence parser for a given language.
5. Construct a recursive descent parser for an expression.
6. Construct a LL(1) parser for an expression
7. Design predictive parser for the given language
8. Implementation of shift reduce parsing algorithm.
9. Convert the BNF rules into Yacc form and write code to generate abstract syntax tree.

Note: This is only a suggested list of lab exercise. Instructor may frame additional exercise relevant to the course contents.

EIGHTH SEMESTER

Course Code: CSE80120	**Course Title:** Major Project Work				
Scheme and Credits:	**L**	**T**	**P**	**Credits**	**Total marks: 300**
Semester: 8th	-	1	10	6	
Course Objectives: • To apply the theoretical knowledge and practical skills acquired in the course of study towards accomplishment of well-defined objectives. • To document the project work and the results accomplished in a professional manner.					

Description:

- In the Major Project, students shall carry out complete implementation of their projects chosen in the pre- project. The main focus shall be towards exposing students to latest technologies/tools having relevance to current market trends.
- The progress of the project work shall be evaluated on a regular basis both by the project guide as well through Departmental presentations.
- Students shall devote time to their project work and consult their project guide/s regularly. The time table shall contain slots exclusively for the same.
- Students who fail to clear the internal project evaluation won’t be eligible for the external project evaluation.
- Students must follow the basic principles of programming such as modularity, clarity and liberal use of comments along with precise documentation.
- The work shall culminate in a project report. Students shall prepare their project reports as per the prescribed format following all guidelines issued in this regard.

Evaluation:

- Out of the 100 marks stipulated for Internal Semester Evaluation (ISE), 50 shall be awarded on the basis of continuous assessment by the respective Supervisor, while the remaining 50 marks shall be evaluated during the project presentation to be held before the Departmental Committee.
- The External Semester Evaluation (ESE) shall be held by an external examiner. The External Semester Evaluation (ESE) shall be of 200 marks. The break-up of 200 marks shall be as follows:

Relevance of work, innovation and future scope: 20 marks.
Presentation: 30 marks
Viva-voce: 100 marks
Report: 50 marks

Course Code: CSE80220	**Course Title:** Industrial Training and Professional Viva				
Scheme and Credits:	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 8th	-	-	2	1	

Objective:

The students are required to go for internship/industrial training of 4 to 6 weeks duration at the end of the 5th or 7th semester after the examinations are over. Internships are educational and career development opportunities, providing practical experience in a field or discipline. They are structured, short-term, supervised placements often focused around particular tasks or projects with defined timescales. An internship will expose computer engineering students to the real IT industry environment, which cannot be simulated in the classroom and hence create competent professionals for the industry. The internship will Expose students to the current technological developments relevant to the subject area of training. It will make students Understand the social, economic and administrate considerations that influence the working environment of IT organisations.

Evaluation through Seminar Presentation/ Viva-Voce:

The student will give a seminar based on training report, before an expert committee constituted by the concerned department and the evaluation be done on the basis of:

- Quality of content presented.
- Proper planning for presentation.
- Effectiveness of presentation.
- Depth of knowledge and skills.

ELECTIVE-I

Course Code: PEC61120	**Course Title:** Advanced Algorithm Design				
Scheme and Credits:	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 6th	3	-	2	4	

Course Objectives: The objectives of course are:

- To analyse the performance of algorithms.
- To choose appropriate data structures and algorithm design methods for a specified application.
- To understand how the choice of data structures and the algorithm design methods impact the performance of programs.

UNIT-I No of Lectures: 14

Introduction: Role of Algorithms in computing, Order Notation, Recurrences, Probabilistic Analysis and Randomized Algorithms. Sorting and Order Statistics: Heap sort, Quick sort and Sorting in Linear Time.

Advanced Design and Analysis Techniques: Dynamic Programming- Matrix chain Multiplication, Longest common Subsequence and optimal binary Search trees.

UNIT-II No of Lectures: 15

Greedy Algorithms: Huffman Codes, Activity Selection Problem. Amortized Analysis. Graph Algorithms: Topological Sorting, Minimum Spanning trees, Single Source Shortest Paths, Maximum Flow algorithms.

Divide and Conquer: Sorting Algorithms, Lower Bounds, Median in Linear time, Maximum Sub-array and Closest Pair of points, Decrease and Conquer variants.

UNIT-III No of Lectures: 15

Sorting Networks: Comparison Networks, Zero-one principle, Bitonic Sorting Networks, Merging Network, Sorting Network. Matrix Operations- Strassen's Matrix Multiplication, Inverting matrices, Solving system of linear Equations.

String Matching: Naive String Matching, Rabin-Karp algorithm, matching with finite Automata, Knuth Morris - Pratt algorithm.

UNIT-IV No of Lectures: 12

NP-Completeness and Approximation Algorithms: Polynomial time, polynomial time verification, NP-Completeness and reducibility, NP-Complete problems.

Approximation Algorithms- Vertex cover Problem, Travelling Sales person problem.

List of Books:

1. Introduction to Algorithms, by Cormen, Leiserson, Rivest, and Stein.
2. Computer Algorithms, by Horowitz, Sahni, and Rajasekaran.
3. Algorithm Design, by Kleinberg and Tardos.
4. Algorithm Design: Foundations, Analysis and Internet examples, M.T. Goodrich and R. Tomassia, John Wiley and sons.
5. Data structures and Algorithm Analysis in C++, Allen Weiss, Second edition, Pearson education.

Course Code: PEC61220	**Course Title:** Software Testing				
Scheme and Credits:	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 6th	3	-	2	4	

Course Objectives:

1. Understanding the role of software testing in software development.
2. Understanding various testing techniques.
3. Understanding Test planning and management
4. Familiarisation with testing of object-oriented systems and web applications.

UNIT-I No. of Lectures: 13

Overview of Software Evolution, SDLC, Testing Process, Terminologies in Testing: Error, Fault, Failure, Verification, Validation, Test Cases, Testing Suite, Impracticality of Testing All Data; Impracticality of Testing All Paths.

Verification Methods, SRS Verification, Source Code Reviews, User Documentation

Verification, Project Audit, Tailoring Software Quality Assurance Program by Reviews, Walkthrough, Inspection and Configuration Audits.

UNIT-II No. of Lectures: 14

Functional Testing: Boundary Value Analysis, Equivalence Class Testing, Decision Table Based Testing, Cause Effect Graphing Technique.

Structural Testing: Control Flow Testing, Path Testing, Independent Paths, Generation of Graph from Program, Identification of Independent Paths, Cyclomatic Complexity, Data Flow Testing, Mutation Testing.

UNIT-II No. of Lectures: 14

Integration, System, and Acceptance Testing: Top down and Bottom up integration, Bi-directional integration, System integration, Scenario Testing, Defect Bash, Functional versus Non-functional testing, Design/Architecture verification, Deployment testing, Beta testing, Scalability testing, Reliability testing, Stress testing, Acceptance testing: Acceptance criteria, test cases selection and execution. Reducing the number of test cases: Prioritization guidelines.

UNIT-III No. of Lectures: 13

Test Planning, Management, Execution and Reporting, Software Test Automation: Scope of automation, Design & Architecture for automation, Generic requirements for test tool framework, Test tool selection, Introduction to Testing in Object Oriented Systems and Web Applications.

Text Book:

1. Desikan and G. Ramesh, "Software Testing: Principles and Practices", Pearson Education.
2. Aditya P. Mathur, "Fundamentals of Software Testing", Pearson Education.
3. Naik and Tripathy, "Software Testing and Quality Assurance", Wiley
4. K. K. Aggarwal and Yogesh Singh, "Software Engineering", New Age International Publication

Course Code: PEC61320	**Course Title:** Numerical Methods				
Scheme and Credits:	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 6th	3	0	2	4	
Course Objectives: The objective of this course is to introduce students to the various numerical techniques, which find their applications in almost every sphere of Science and Engineering and are capable of handling large systems of equations, non-linearities and complicated geometries that are common in engineering practice but often impossible to solve analytically. The course emphasizes algorithm on development and programming and application to realistic engineering problems.					

UNIT-I No. of Lectures: 14

Solution for linear and nonlinear equations:

Algebraic equations: Bisection method. Regula-Falsi method. Newton Raphson Method. Secant method.

Solution for systems of equations:

Basic Gauss Elimination method, Gauss Elimination with Pivoting, Gauss Jordan method for linear system of equations.

UNIT-II No. of Lectures: 14

Interpolation:

Meaning of Δ, E, μ and δ. Forward difference, Backward difference, and Central difference, Newton's formula. Lagrange's Interpolation formula. Sterling's and Bessel's formula.

UNIT-III No. of Lectures: 14

Numerical Differentiation and Integration:

Derivations from difference tables, Higher order derivations. Newton-Cotes Methods: Trapezoidal rule, Simpson's 1/3 rule, Simpson's 3/8 rule. Gaussian Integration.

Difference Equations and their solutions:

Single-step methods: Taylor series method, Euler's method, Runge- Kutta methods of 2nd and 4th order. Multi-step methods: Miline's Predictor Corrector formulas, Adam-Bashforth formula.

UNIT-IV No. of Lectures: 14

Partial Differential Equations:

Finite difference techniques for the solution of two-dimensional Laplace's and Poison's equations on rectangular domain – One dimensional heat flow equation by explicit and implicit (Crank Nicholson) methods – One dimensional wave equation by explicit method.

List of Books:

1. Numerical Methods, E. Balaguruswamy, TMH.
2. Introduction to Numerical Methods, V. Rajaraman, TMH.
3. Introduction to Numerical Analysis, Froberg C. E., 2nd edition, Addison Wesley.
4. Applied Numerical Analysis, Gerald C. F., Wheatley P.O., 6th edition, Pearson Asia.
5. Applied Numerical Methods for Engineers using MATLAB and C, Schilling, Cengage India.
6. Numerical Mathematics and Computing, Cheney, Cengage India.
7. Numerical methods for engineers by Chapra and Canale, 5th edition, Tata McGraw Hill.

Course Code: PEC61420	**Course Title:** Embedded Systems				
Scheme and Credits:	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 6th	3	0	2	4	
Course Objectives: The objective of the course is to make students understand the basics of embedded systems. It will enable the students to acquire knowledge and understand fundamental embedded systems design paradigms and architectures. After going through this course, the students shall be able to practically apply gained theoretical knowledge to design, analyse and implement embedded systems.					

UNIT-I No of Lectures: 14

Fundamentals of Embedded System: Core of the embedded system, Memory, Sensors (resistive, optical, position, thermal) and Actuators (solenoid valves, relay/switch, opto-couplers), Communication Interface, Embedded firmware (RTOS, Drivers, Application programs), Power-supply (Battery technology, Solar), PCB and Passive components, Safety and reliability, environmental issues. Ethical practice.

Characteristics and quality attributes (Design Metric) of embedded system. Real time system's requirements, real time issues, interrupt latency.

Embedded Product development life cycle, Program modelling concepts- DFG, FSM, Petri-net.

UNIT-II No of Lectures: 12

Embedded Hardware and Design:

Introduction to ARM-v7-M (Cortex-M3), ARM-v7-R (CortexR4) and comparison in between them.

UNIT-III No of Lectures: 14

Embedded Serial Communication: Study of basic communication protocols like SPI, SCI (RS232, RS485), I2C, 10 CAN, Field-bus (Profibus), USB (v2.0), Bluetooth, Zig-Bee, Wireless sensor network.

UNIT-IV No of Lectures: 14

Embedded Software, Firmware Concepts and Design: Embedded C-programming concepts (from embedded system point of view).

Real time operating system: Need of RTOS in Embedded system software, RTOS services in contrast with traditional OS.

List of Books:

1. Introduction to Embedded Systems: Shibu K. V. (TMH)
2. Embedded System Design – A unified hardware and software introduction: F. Vahid (John Wiley)
3. Embedded Systems: Rajkamal (TMH)
4. Embedded System design: S. Heath (Elsevier)
5. Embedded Microcontroller and processor design: G. Osborn (Pearson)
6. Embedded Systems: Frank Vahid , Wiley India, 2002
7. Embedded Microcomputer Systems – Real Time Interfacing – Jonathan W. Valvano; Cengage Learning; Third or later edition.

Course Code: PEC61520	**Course Title:** Data Mining and Warehousing				
Scheme and Credits:	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 6th	3	0	2	4	

Course Objectives:

This course helps the students to understand the overall architecture of a data warehouse and methods for data gathering and data pre-processing using OLAP tools. The different data mining models and techniques will be discussed in this course. The course aims to teach the basic principles, concepts and applications of data warehousing and data mining, introduces the task of data mining as an important phase of knowledge recovery process and familiarizes Conceptual, Logical, and Physical design of Data Warehouses. It also imparts knowledge of the fundamental concepts that provide the foundation of data mining.

UNIT-I No. of Lectures: 14

Data Warehousing & modelling: Basic Concepts, Data Warehousing- A multitier Architecture, Data warehouse models: Enterprise warehouse, Data mart and virtual warehouse, Extraction, Transformation and loading, Data Cube: A multidimensional data model, Stars, Snowflakes and Fact constellations: Schemas for multidimensional Data models, Dimensions: The role of concept Hierarchies, Measures: Their Categorization and computation, Typical OLAP Operations.

UNIT-II No. of Lectures: 12

Data Warehouse implementation: Efficient Data Cube computation: An overview, Indexing OLAP Data: Bitmap index and join index, Efficient processing of OLAP Queries, OLAP server Architecture ROLAP versus MOLAP.

UNIT-III No. of Lectures: 14

Data Mining: Data Mining Functionalities, Data Pre-processing, Data Cleaning, Data Integration and Transformation, Data Reduction, Data Discretization and Concept, Hierarchy Generation, Architecture of a typical Data Mining system, Classification of Data Mining systems.

Association Rule Mining: Efficient and Scalable Frequent Item set Mining Methods, Mining various kinds of Association Rules, Association Mining to Correlation Analysis, Constraint-Based Association Mining.

UNIT-IV No. of Lectures: 14

Classification: Issues Regarding Classification and Prediction, Classification by Decision Tree Introduction, Bayesian Classification, Rule Based Classification, Classification by Back propagation, Support Vector Machines, Associative Classification, Lazy Learners, Other Classification Methods

List of Books:

1. Jiawei Han, Micheline Kamber, Jian Pei (2012), Data Mining: Concepts and Techniques, 3rd edition, Elsevier, United States of America.
2. Margaret H Dunham (2006), Data Mining Introductory and Advanced Topics, 2nd edition, Pearson Education, New Delhi, India.
3. Amitesh Sinha (2007), Data Warehousing, Thomson Learning, India.
4. Xingdong Wu, Vipin Kumar (2009), the Top Ten Algorithms in Data Mining, CRC Press, UK.
5. Alex Berson and Stephen J. Smith “Data Warehousing, Data Mining & OLAP”, Tata McGraw – Hill Edition, Tenth Reprint 2007.
6. G. K. Gupta “Introduction to Data Mining with Case Studies”, Easter Economy Edition, Prentice Hall of India, 2006.
7. Pang-Ning Tan, Michael Steinbach and Vipin Kumar “Introduction to Data Mining”, Pearson Education, 2007.

Course Code: PEC61620	**Course Title:** Enterprise Java				
Scheme and Credits:	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 6th	3	-	2	4	
Course Objectives: To enable students to understand the concepts underlying technologies in Java Enterprise edition, Configuring web/application servers, Java beans and Enterprise Java Beans.					

UNIT-I No of Lectures: 14

Understanding Java EE: Enterprise Application, Java enterprise edition, Java EE Technologies, Java EE Architecture. Server and Containers: Types of System Architecture, Java EE Server, Java EE Containers. **Introduction to Java Servlets:** The Need for Dynamic Content, Java Servlet Technology. Servlet API and Lifecycle: Java Servlet API, The Servlet Skeleton, The Servlet Life Cycle. Using Annotations Instead of Deployment Descriptor. Working with Databases: JDBC Architecture, Accessing Database, The Servlet GUI and Database Example.

UNIT-II No of Lectures: 14

Request Dispatcher: Requestdispatcher Interface, Methods of Requestdispatcher, Requestdispatcher Application. Cookies: Kinds of Cookies, Creating Cookies Using Servlet. Session: Lifecycle of Http Session, Session Tracking with Servlet API, A Servlet Session Example.

Working with Files: Uploading Files, Creating an Upload File Application, Downloading Files, Creating a Download File Application. Working with Non-Blocking I/O: Creating a Non-Blocking Read Application

UNIT-III No of Lectures: 12

Java Server Pages: Java Server Pages, Use, Merits, Life Cycle of a JSP Page, Comments, JSP Document, JSP Elements, JSP GUI Example. Action Elements; Implicit Objects, Scope and El Expressions; Java Server Pages Standard Tag Libraries.

UNIT-IV No of Lectures: 16

Introduction to Enterprise Javabeans: Enterprise Bean Architecture: Benefits of Enterprise Bean, Types of Enterprise Bean, Accessing Enterprise Beans, Enterprise Bean Application, Packaging. Working with Session Beans: Example of Stateful Session Bean, Example of Stateless Session Bean, Example of Singleton Session Beans. Working with Message Driven Beans: Lifecycle of a Message Driven Bean, Uses of Message Driven Beans, The Message Driven Beans Example.

Introduction to Hibernate: Writing hibernate applications.

List of Books:

1. Black Book “Java server programming” J2EE, Dream Tech Publishers, Kathy Walrath.
2. Complete Reference J2EE by James Keogh mcgraw publication.
3. Deitel & Deitel, “Java How to program”, Prentice Hall.
4. Professional Java Server Programming by Subrahmanyam Allamaraju, Cedric Buest Wiley Publication.
5. Beginning JSP, JSF and Tomcat, Giulio Zambon, Apress

ELECTIVE-II

&

ELECTIVE-III

Course Code: PEC71120	**Course Title:** Machine Learning				
Scheme and Credits:	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 7th	3	0	2	4	

Course Objectives: The course objectives are:

- Understanding the fundamental issues and challenges of machine learning: data, model selection, model complexity etc.
- Examine meaningful features from a given dataset by learning pre-processing skills.
- Propose and implement various machine learning algorithms in a range of real-world applications.
- Apply the model evaluation and hyper parameter tuning methods to enhance the modelaccuracy.

UNIT-I No. of Lectures: 14

Giving computers the ability to learn from data: Building intelligent machines to transform data into knowledge, different types of machine learning, roadmap for building machine learning systems, using Python for machine learning

Building good training sets: Data pre-processing, dealing with missing data, handling categorical data, partitioning a dataset in training and test sets, bringing features onto the same scale, selecting meaningful features

UNIT-II No. of Lectures: 14

Machine learning classifiers: choosing a classification algorithm, implementing perceptron learning, modelling class probabilities via logistic regression, maximum margin classification with support vector machine, decision tree learning, k-nearest neighbour algorithm, Bayesian learning, combining different models for Ensemble learning, majority voting classifier, bagging and boosting classifier, random forest classifier.

UNIT-III No. of Lectures: 14

Predicting continuous target variables with regression analysis: introducing linear regression, evaluating the performance of linear regression model, relationship using a correlation matrix, exploratory data analysis, regularized methods for regression, polynomial regression, modelling nonlinear relationships in the housing dataset, decision tree and random forest regressor

UNIT-IV No. of Lectures: 14

Working with unlabelled data - Clustering Analysis: grouping objects by similarity using K-means clustering, hard versus soft clustering, using the elbow method to find the optimal number of clusters, organizing clusters as a hierarchical tree.

Compressing data via dimensionality reduction: Unsupervised dimensionality reduction via principal component analysis, supervised data compression via linear discriminant analysis, using kernel principle component analysis for nonlinear mappings, projecting new data points

Model evaluation and hyper parameter tuning: Using k-fold cross validation to access model performance, looking at different performance evaluation metrics.

List of Books:

1. Shalev- Shwartz,S., Ben-David,S., (2014), Understanding Machine Learning: From Theory to Algorithms, Cambridge University Press.
2. R. O. Duda, P. E. Hart, D. G. Stork (2000), Pattern Classification, Wiley-Blackwell, 2nd Edition
3. Python machine learning by sebastian raschka, packt publishing.
4. Mitchell Tom (1997). Machine Learning, Tata McGraw-Hill.
5. Learning scikit-learn: machine learning in python by raul garreta, guillermo moncecchi, packt publishing.

Course Code: PEC71220	**Course Title:** Software Project Management				
Scheme and Credits:	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 7th	3	0	2	4	

Course Objectives: Through this course students should be able to:

- Describe basics of software project management and step wise project planning.
- Apply cost and effort estimation techniques on software projects.
- Apply methods and techniques of activity planning and risk assessment on software projects.
- Analyse the progress of software project using monitoring and controlling tools.

UNIT-I No of Lectures: 14

Introduction to Software project management: Software project versus other types,Nature of software production, Activities by Software Project Management, Key objectives of effectivemanagement, Problems with software projects, Risk reduction, Meaning of project, Software ProjectVersus Other Types, Nature of software production, Importance of software project management.

Stepwise Project planning: Project scope, Objectives, Business Planning: determining objectives, Infrastructure, forecasting demand for product, Characteristics, Proposal writing, Effort estimation, Requirement analysis, Risk identification, Legal issues, Risk identification.

UNIT-II No of Lectures: 14

Cost Estimation and Life Cycle Models: Meaning, managing allocation of resources, Creating programme, Individual projects, Program management, Risk evaluation, Cost benefit analysis, Evaluation of individual projects, Introduction to the life cycle models.

Effort Estimation: Meaning, Problems with Estimation Basis, Estimation Techniques Albrecht Function Point Analysis, Functions Mark II, COCOMO Model, COCOMO extensions.

UNIT-III No of Lectures: 14

Activity Planning: Objectives, Project Schedule.\, Managing the task, managing the plan, Network Planning Model, Time Dimension, Identifying Critical Path, managing change, readjusting goals and milestones.

Risk Management: Risk, Categories of Risk, Dealing with Risk, Risk Identification, Risk Assessment, Risk Planning, Risk Management, Evaluating Risk to the Schedule, Applying the PERT Technique

UNIT-IV No of Lectures: 14

Resource Allocation, Monitoring and Control: Resource allocation introduction, identifying resource requirements, Scheduling resources, Resource allocation, Publishing the resource & cost schedule, scheduling sequence, creating frameworks, Data collection, Visualizing progress, Status reports, Milestone analysis, Cost monitoring, change control, Cost (direct and indirect), Earned value analysis, Performance ratio.

Software quality & small projects: Introduction, Defining software quality., Software Quality: IS09126, Software measures, Product versus process quality., Management of external standards,Problems with student projects, Content of project plan.

List of Books:

1. Software Project Management by Bob hughes, Mike cotterell, Rajibmall, McGraw hill education.
2. Software Project Management By Mohapatra Dr Sanjay, Cengage Learning.
3. Software Project Management: A Real-World Guide To Success By JoelHenry, Pearson.
4. Software Project Management By Subramanian Chandramouli, SaikatDutt, Pearson.

Course Code: PEC71320	**Course Title:** Digital Image Processing				
Scheme and Credits:	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 7th	3	-	2	4	

Course Objectives: The course objectives are:

- To understand different techniques employed for the enhancement of images.
- To learn different causes for image degradation and overview of image restoration techniques.
- To understand the need for image compression and to learn the spatial and frequency domain techniques of image compression.
- To learn different feature extraction techniques for image analysis and recognition.

UNIT-I No of Lectures: 14

Digital Image Fundamentals: Fundamentals of image processing, applications of image processing, elements of visual perception, image sampling and quantization, basic relationship between pixels, basic geometric transformation.

Transformations: Introduction to Fourier Transform, properties of 2D Fourier Transform, FFT, separable image transforms-Walsh, Hadamard, Discrete Cosine Transform and Haar Transform.

UNIT-II No of Lectures: 14

Image Enhancement: Spatial domain methods- Basic grey level transformation, Histogram equalization, Image subtraction, Image averaging. Spatial filtering: Smoothing, sharpening filters,Laplacian filters. Frequency domain filters: Smoothing, Sharpening filters. Homomorphic filtering.

UNIT-III No of Lectures: 14

Image Degradation and Restoration: Model of Image degradation and restoration process, noise models, inverse filtering, Least mean square filtering, constrained least mean square filtering, blind image restoration, pseudo inverse, singular value decomposition.

UNIT-IV No of Lectures: 14

Image Compression: Lossless Compression- Variable length coding, LZW coding, bit plane coding, predictive coding. Lossy Compression- Transform coding, wavelet coding. Basic image compression standards- JPEG, MPEG.

Edge Detection and Segmentation: Edge detection, thresholding, region-based segmentation Boundary representation: chair codes- Polygonal approximation, boundary segments, boundary descriptors: simple descriptors Fourier descriptors, regional descriptors, simple descriptors, texture.

List of Books:

1. Digital Image Processing By Rafael C. Gonzalez, Richard Eugene Woods.
2. Fundamentals of Image Processing by Anil K. Jain Prentice Hall.
3. Kenneth R. Castleman, Digital Image Processing ‘, Pearson, 2006
4. Rafael C. Gonzalez, Richard E. Woods, Steven Eddins, Digital Image Processing using MATLAB, Pearson Education, Inc., 2011.
5. William K. Pratt, Digital Image Processing ‘, John Wiley, New York, 2002

Course Code: PEC71420	**Course Title:** Internet of Things				
Scheme and Credits-	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 7th	3	0	2	4	

Course Objectives: The course objectives are:

- Understand the basic characteristics of IoT.
- Identify the new models for market strategic interaction.
- Review various communication components of IoT.
- Understand IoT Architecture and Protocols.
- Understand the privacy and security issues.
- Identify future trends of IoT in Business.

UNIT-I No of Lectures: 12

IoT & Web Technology: Introduction to IoT, Concept and Terminology of the Internet of Things, History of IoT, Requirements of IoT, Internet of Things Vision, Challenges in IoT, IoT Strategic Research and Innovation Directions, Future Internet Technologies, Security, Privacy & Trust

UNIT-II No of Lectures: 12

M2M to IoT: Introduction, Some Definitions, M2M Value Chains, IoT Value Chains, An emerging industrial structure for IoT, IoT enabling technologies- Sensors, Actuators, Gateways, Local and Global connectivity, Introduction to wireless communication network, MANET

UNIT-III No of Lectures: 14

IoT -State of the Art: Reference Architecture Model- Introduction, IoT reference Model, IoT Protocols Functional View, Information View, Deployment and Operational View, Other Relevant architectural views.

Security and Governance: Introduction, Overview of Governance, Privacy and Security Issues, Contribution from FP7 Projects, Security, Privacy and Trust in IoT-Data-Platforms for Smart Cities

UNIT-IV No of Lectures: 16

IoT Applications for Value Creations: Introduction, IoT applications for industry: Future Factory Concepts, Brownfield IoT, Smart Objects, Smart Applications

IoT In Industry: Four Aspects in your Business to Master IoT, IoT for Retailing Industry, IoT For Oil and Gas Industry

Developing IoTs: Introduction to Python, Introduction to different IoT tools, developing applications through IoT tools, developing sensor-based application through embedded system platform, Implementing IoT concepts with python

List of Books:

1. Vijay Madisetti, Arshdeep Bahga, Internet of Things: A Hands-On Approach, Orient Blackswan Pvt. Ltd.- New Delhi, First Edition, 2015.
2. Waltenegus Dargie, Christian Poellabauer, Fundamentals of Wireless Sensor Networks: Theory and Practice, Wiley-Blackwell, July 2010.
3. Francis da Costa, Rethinking the Internet of Things: A Scalable Approach to Connecting Everything, 1st Edition, Apress Publications, 2013
4. Jan Holler, Vlasios Tsiatsis, Catherine Mulligan, Stefan Avesand, Stamatis Karnouskos, David Boyle, From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence, 1st Edition, Academic Press, 2014.

Course Code: PEC71520	**Course Title:** Introduction to Big Data				
Scheme and Credits:	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 7th	3	-	2	4	

Course Objectives: The course objectives are:

- Analyse the need and importance of fundamental concepts of Big Data.
- Understand internal functioning of different modules of Apache Hadoop.
- Evaluate performance of Big Data problems using Map Reduce model.
- Infer results of real time applications using Apache Flume.

UNIT-I No. of Lectures: 14

Introduction to Big Data Platform: Challenges of Conventional Systems, Intelligent data analysis, Nature of Data, Analytic Processes and Tools, Analysis vs Reporting, Modern Data Analytic Tools, Statistical Concepts: Sampling Distributions, Re-Sampling, Statistical Inference, Prediction Error Mining data streams: Introduction to Streams Concepts, Stream Data Model and Architecture, Stream Computing

UNIT-II No. of Lectures: 14

Introduction to Hadoop: Basics of Hadoop, Ubuntu Linux Installation, Hadoop Installation in Standalone Mode, Hadoop Installation in Single Node Cluster Mode, Configuring SSH for Hadoop, Configuring HDFS, Formatting and Starting the Single Node Cluster, Various Hadoop Commands

UNIT-II No. of Lectures: 14

Hadoop Architecture: Hadoop Core Components, Daemons of Hadoop and their functionality-Name Node, Data Node, Secondary Name Node, Job Tracker, Task Tracker, Hadoop Architecture

Map Reduce Model for Hadoop: Introduction to Map Reduce, MapReduce Terminology, Hadoop MapReduce paradigm, Steps for Writing Map Reduce Program, Map Reduce Design Patterns, Map Reduce Programs

UNIT-III No. of Lectures: 14

Hadoop Streaming with Apache Flume: Installation of Apache Flume- Environment and Configuration, Architecture of Apache Flume- Flume Event, Flume Agent, Flume Data Flow process, Hadoop Streaming, Fetching Twitter Data in HDFS

Introduction to Apache Hive: Hive Installation, Hive Data Types, Hive Partitioning, Hive Operators, Hive Functions, Hive Views and Indexes, HiveQL Operations

Text Book:

1. Michael Berthold, David J. Hand, “Intelligent Data Analysis”, Springer, 2007.
2. Tom White “Hadoop: The Definitive Guide” Third Edition, O’reilly Media, 2012.
3. Anand Rajaraman and Jeffrey David Ullman, “Mining of Massive Datasets”, CUP, 2012.
4. Bill Franks, “Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics”, John Wiley& sons, 2012.
5. Glenn J. Myatt, “Making Sense of Data”, John Wiley & Sons, 2007.

Course Code: PEC71620	**Course Title:** Advanced Computer Networks				
Scheme and Credits:	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 7th	3	0	2	4	
Course Objectives: The course objectives are: • Understanding the state-of-the-art WAN technologies. • Understanding of routing algorithms and routing metrics. • Understanding QoS and Network Management.					

UNIT-I No. of Lectures: 14

Asynchronous transfer mode: ATM protocol architecture, ATM logical connection, ATM cell, ATM service categories, AAL.

MPLS: Benefits of MPLS, MPLS architecture, forwarding labelled packets, LDP overview.

UNIT-II No. of Lectures: 14

Routing: Overview of routing algorithms, features of a routing algorithm, classification, shortest path algorithm, Dijkstra algorithm, distance vector and link state algorithm, routing between peers (BGP)

Multicast: IGMP, PIM, DVMRP, mobility, mobile IP.

UNIT-III No. of Lectures: 12

Flow and Congestion Control: Window and rate-based schemes, decbit. ATM-ABR;

Quality of Service: IETF integrated services model, differentiated services model.

UNIT-IV No. of Lectures: 14

Traffic Flows: Flow identification, packet classifiers and filters.

Network Management: SNMP, CMIP. Issues in the management of large networks.

Programmable Networks: Overview of programmable networks, SDN, OpenFlow, Network Virtualization.

List of Books:

1. J. F. Kurose “Computer Networking” Addison-Wesley.
2. Douglas E. Comer, “Internetworking with TCP/IP, Volume 1”, PHI.
3. Douglas E. Comer, “Internetworking with TCP/IP, Volume 2”, PHI.
4. L. D. Ghein, “MPLS Fundamentals”, CISCO.
5. T. D. Nadeau and K. Gray, “Software Defined Networks”, O’REILLY.
6. Douglas E. Comer, “Client-Server Programming with TCP/IP, Volume3”, PHI.

Course Code: PEC71720	**Course Title:** Android Programming				
Scheme and Credits:	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 7th	3	0	2	4	

Course Objectives:

This course facilitates classroom and laboratory learning, letting students develop competence and confidence in android programming and understand the entire Android Apps Development Cycle, as well as it would also enable the students to independently create Android Applications.

UNIT-I No. of Lectures: 12

Mobile Applications, Characteristics and Benefits, Application Model, Infrastructure and Managing Resources.

Mobile Software Engineering: Frameworks and Tools, Mobile devices Profiles. Android Architecture, editing Emulator settings, launching Emulator, Android API Levels, setting up development environment, Dalvik Virtual Machine, Activities and Services.

UNIT-II No. of Lectures: 15

Understanding units and layout, Using layout managers.

Creating Activities, Android manifest.xml, Resources and R.java, Textview, Button,

Basic Controls: edittext, Radiobutton, Checkbox, Array and Base Adapters, Listview, Gridview using adapters.

Creating and using styles; Creating and using themes, creating icons, Creating NinePatch drawables, Setting up frame-by-frame animation.

UNIT-III No. of Lectures: 15

Android Intents, working with implicit and explicit Intents, linking Activities using Intents, passing data using an Intent object, returning results from Intent

Menus and views, working with Option Menu**,** sub-menu and context menu**,** using List Views to display long lists**,** using Image Views to display pictures**,** displaying pictures and Menus with Views. Understanding screen size and density; Providing alternate layouts.

UNIT-IV No. of Lectures: 14

Creating and using databases**,** update, Database operations: insert, delete

Shared Preferences, App Setting, SQLite primer, Store data using SQLite database, Content Providers, Content Resolver, Loader.

Performance Improvement of App: Performance Parameters, Profiling Tools, Rendering and Layout, Garbage Collection and Memory Leaks, Best Practices.

Testing Android applications, Publishing Android application, Using Android preferences, Managing Application resources in a hierarchy, Working with different types of resources.

List of Books:

1. Beginning android programming with android studio by J. F. Dimarzio, Wiley.
2. Beginning android programming: develop and design by Chris Haseman, Pearson Education India.
3. Android programming unleashed by B.M Harwani, Pearson Education India.
4. Hello, Android: Introducing Google's Mobile Development Platform by Ed Burnett.

<table>
<tr><td><b>Course Code:</b> PEC71820</td><td colspan="5"><b>Course Title:</b> Full stack development-I</td></tr>
<tr><td><b>Scheme and Credits:</b></td><td><b>L</b></td><td><b>T</b></td><td><b>P</b></td><td><b>Credits</b></td><td rowspan="2"><b>Total marks: 100 marks</b></td></tr>
<tr><td><b>Semester:</b> 7th</td><td>3</td><td>0</td><td>2</td><td>4</td></tr>
</table>

Course Objectives:

- Understand the fundamentals of Angular Forms and its architecture
- Present data in beautiful, interactive lists
- Build forms and setting pages
- Implement Single page application (SPA)

UNIT-I No. of Lectures: 14

Introduction: Introduction to full stack development, importance, requirements, development tools.

Angular JS Basics: Angular JS, MVC, MVC-The Angular JS way, Features of Angular JS, Model-View-Controller. Angular expressions, how to use expressions, Angular vs JavaScript.

UNIT-II No. of Lectures: 14

Expressions and Data Biding: Number and String Expressions, Object Binding and Expressions, Working with Arrays, Forgiving Behaviour, Understanding Data binding

Working with Directives: Conditional Directives, Styles Directives, Mouse and Keyboard Events Directives

Controllers: Understanding Controllers: Programming Controllers & $scope object, Adding Behaviour to a Scope Object, Passing Parameters to the Methods, Having Array as members in Controller Scope., Nested Controllers and Scope Inheritance, Multiple Controllers and their scopes.

UNIT-III No. of Lectures: 14

Filters: Built-In Filters, Uppercase and Lowercase Filters, Currency and Number Formatting Filters, OrderBy Filter, Filter Filter, Creating Custom Filter.

Forms: Using Simple Form, Working with Select and Options, Input Validations, Using CSSclasses, Form Events, Custom Model update triggers, Custom Validations.

Modules: Module Loading and Dependencies, Recommended Setup of Application, Creation vs Retrieval.

UNIT-IV No. of Lectures: 14

Services: Understanding Services, Developing Creating Services, Using a Service, Injecting Dependencies in a Service.

Ajax in Angular JS: $http Service, $q Service, $http and $q Service

Routing: Introduction to SPA (Single page application), Creating HTML Templates,Configuring Route Provider.

List of Books:

1. Krishna Rungta , “Learn AngularJS in 1 Day: Complete Angular JS Guide with Examples”
2. Asim Hussain “Angular: From Theory To Practice”
3. Matt Frisbie “Angular 2 Cookbook”

Course Code: PEC71920	**Course Title:** Introduction to Data Science				
Scheme and Credits:	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 7th	3	0	2	4	

Course Objectives:

1. Apply aptitude for business improvement, innovation and entrepreneurial action.
2. Apply basic Excel concepts in solving the business-related problems.
3. Identify, model and solve decision problems in different settings.
4. Explain how data is collected, managed and stored for data science.
5. Critically evaluate data visualizations based on their design and use for communicating stories from data.

UNIT-I No. of Lectures: 14

Introduction: Data science, Terminology, The data science process, data science toolkit, Types of data, applications.

Data Collection and Management: Sources of data, Data collection and APIs, Exploring and fixing data, Data storage and management, Using multiple data sources.

UNIT-II No. of Lectures: 14

Descriptive Analysis: Descriptive Statistical measures - Populations and samples, Statistical notations, Measures of Location, Measures of Dispersion, and Measures of Association.

Statistical Inference: Hypothesis testing, one-Sample Test, Two-Sample Test, Two tailed Hypothesis for mean, ANOVA.

Analytics on Spreadsheets: Basic Excel, Excel Formulas, Excel Functions, Data Queries

UNIT-III No. of Lectures: 14

Predictive Analytics: Simple Linear regression, Multiple Linear regression, Residual Analysis, Building regression models, Regression with Categorical Independent variables – case studies**.**

Prescriptive Analytics: Building Linear Optimization models, Implementing Linear Optimization models on spreadsheets, Solving Linear Optimization models- case studies.

UNIT-IV No. of Lectures: 14

Basic machine learning algorithms: Linear regression, SVM, Naive Bayes

Data Visualisation: Introduction, Exploratory and Explanatory data visualization, data types, Data encodings, Retinal variables, Mapping variables to encodings, Visual encodings, Technologies for visualisation.

List of Books:

1. James Evan, Business Analytics- Methods, Models, and Decisions (2nd Edition).
2. Pearson, 2013 Gert H. N. Laursen, Business Analytics for Managers: Taking Business Intelligence Beyond Reporting, Wiley (2nd Edition),2010.
3. S. Christian Albright and Wayne L. Winston, Analytics: Data Analysis and Decision Making, Sixth Edition, 2014.
4. Grus, Joel. Data science from scratch: first principles with python. O'Reilly Media, 2019.

Course Code: PEC72020	**Course Title:** Digital Signal Processing				
Scheme and Credits:	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 7th	3	-	2	4	

Course objective:

- To develop methods for processing discrete-time signals.
- To acquire some familiarity with digital filters in terms design and implementation and to become familiar with how various types of filters affect signal characteristics.
- To become familiar with some applications of digital processing.

UNIT I No. of Lectures: 14

Discrete Time Signals: Introduction to Digital Signal Processing, Representation of discrete time signal.
Systems: classifications of discrete time system, basic operations on sequence, linear, Time invariant, causal, stable.
Linear Time invariant systems: circular convolution, solution to difference equation
Discrete time Fourier series and transform: Discrete time Fourier transform and its properties, Discrete Fourier transform – properties -Fast Fourier transform, Z-transform and its properties.

UNIT II No. of Lectures: 15

Digital filter Structure: Block diagram and signal flow graph representation, structure realization for IIR – Direct form, cascade form, parallel form, lattice structure form, structure realization for FIR – Direct form, cascade form, frequency sampling structure, Lattice structure.
Filters: FIR Filter – windowing technique, optimum equiripple linear phase FIR filter, IIR filter – Butterworth filter, chebyshev filter.

UNIT III No. of Lectures: 15

Multistage Representation: Sampling of band pass signal, anti-aliasing filter, Decimation by an integer factor, interpolation by an integer factor, sampling rate conversion, implementation of digital filter banks, sub-band coding, Quadrature mirror filter.

UNIT IV No. of Lectures: 12

Digital Signal Processors: Fundamentals of fixed-point DSP architecture, Fundamentals of floating- point DSP architecture.
Digital Image processing: Introduction to Digital Image processing, components of image processing system, fundamental steps in image processing, a simple image formation model, image sampling and quantization, basic relationship between pixels.

List of Books:

1. John G. Proakis, Dimitris, G. Manolakis, “Digital Signal Processing: Principles, Algorithms and Applications”, PHI.
2. S.Salivahanan, A.Vallavaraj and C. Gnanapriya, “Digital Signal Processing”, TMH, 2000.
3. A.V. Oppenheim and R.W.Schafer, Englewood, “Digital Signal Processing”, Prentice-Hall Inc, 1975.
4. B.Venkatramani & M.Bhaskar, “Digital Signal Processors architecture, programming and applications”,TMH, 2002.
5. Digital Image Processing, Rafael C Gonzalez, Richard E.Woods, Second Edition, Pearson Education/PHI

ELECTIVE-IV

&

ELECTIVE-V

Course Code: PEC81120	**Course Title:** Introduction to Deep Learning.				
Scheme and Credits:	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 8th	3	0	2	4	

Course Objectives:

- Identify the deep learning algorithms which are more appropriate for various types of learning tasks in various domains.
- Implement deep learning algorithms and solve real-world problems.
- Introduce major deep learning algorithms, the problem settings, and their applications to solve real world problems.

UNIT-I No. of Lectures: 14

Introduction to Deep learning: Deep learning and its applications, intuitions of neural network, machine learning vs deep learning, perspective and issues in learning deep learning framework. **Optimization**: Types of errors, bias-variance trade-off, overfitting, underfitting, brief review of concepts from optimization, variants of gradient descent, momentum-based methods.

UNIT-II No. of Lectures: 14

Artificial Neural Networks: Linear and Logistic Regression. Basic concepts of artificial neurons, single and multilayer perceptron, perceptron learning algorithm, its convergence proof, different activation functions.

ConvNets: Basic concepts of Convolutional Neural Networks. Convolution and pooling operation. ConvNet Architectures: Discussions on famous convnet architectures- AlexNet, VGG, GoogLeNet, ResNet. Discussion on regularization, Dropout, Batchnorm.

UNIT-III No. of Lectures: 14

Recurrent Neural Network (RNN): Sequence learning with neural nets, unrolling the recurrence, training RNN-Back propagation through time (BPTT), Long short-term memory (LSTM), Bidirectional LSTM.

Introduction to Unsupervised Training of Neural Networks.

UNIT-IV No. of Lectures: 14

NLP Basics and Transformers: word2vec, Transformers and Applications to NLP.

Explainability and Bias: Discussion on explainability and bias in Deep Learning system. The need for explanation, introspection vs justification, activation maximization and activation map based explanation generation.

List of Books:

1. Deng & Yu, Deep Learning: Methods and Applications, Now Publishers, 2013.
2. Ian Goodfellow, Yoshua Bengio, Aaron Courville, Deep Learning, MIT Press, 2016.
3. Michael Nielsen, Neural Networks and Deep Learning, Determination Press, 2015.
4. Hands–On Machine Learning with Scikit–Learn and TensorFlow 2e: Concepts, Tools, and Techniques to Build Intelligent Systems by Aurelien Geron.

Course Code: PEC81220	**Course Title:** Introduction to Robotics				
Scheme and Credits:	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 8th	3	0	2	4	

Course Objectives: The course objectives are:

- To introduce the functional elements of Robotics.
- To impart knowledge on the direct and inverse kinematics.
- To introduce the manipulator differential motion and control.
- To educate on various path planning techniques.
- To introduce the dynamics and control of manipulators.

UNIT-I No. of Lectures: 14

Basic Concepts: Brief History, Types of Robot, Technology, Robot classifications and specifications, Design and control issues. Various manipulators: Sensors, work cell, Programming languages.

UNIT-II No. of Lectures: 14

Direct and Inverse Kinematics: Mathematical representation of Robots, Position and orientation, Homogeneous Transformation, Various joints, Representation using the Denavit Hattenberg parameters, Degrees of freedom, Direct kinematics, Inverse kinematics, SCARA robots, Solvability, Solution methods-Closed form solution**.**

UNIT-III No. of Lectures: 14

Manipulator Differential Motion and Statics: Linear and angular velocities, Manipulator Jacobian, Prismatic and rotary joints, Inverse, Wrist and arm singularity, Static analysis, Force and moment Balance.

UNIT-IV No. of Lectures: 14

Path Planning: Definition, Joint space technique, Use of p-degree polynomial, Cubic polynomial, Cartesian space technique, Parametric descriptions, Straight line and circular paths, Position and orientation planning.

Dynamics and Control: Lagrangian mechanics-2DOF, Manipulator-Lagrange Euler formulation, Dynamic model, Manipulator control problem, Linear control schemes, PID control scheme, Force control of robotic manipulator.

List of Books:

1. R.K.Mittal and I.J.Nagrath, Robotics and Control, Tata McGraw Hill, New Delhi,4th Reprint, 2005.
2. John J.Craig , Introduction to Robotics Mechanics and Control, Third edition, Pearson Education,
3. M.P.Groover, M.Weiss, R.N. Nageland N. G.Odrej, Industrial Robotics, McGraw-Hill Singapore, 1996.
4. Ashitava Ghoshal, Robotics-Fundamental Concepts and Analysis', Oxford University Press, Sixth impression, 2010.
5. B.K.Ghosh, Control in Robotics and Automation: Sensor Based Integration, Allied Publishers,Chennai, 1998
6. S.Ghoshal, " Embedded Systems & Robotics" – Projects using the 8051 Microcontroller", Cengage Learning, 2009.

Course Code: PEC81320	**Course Title:** Cloud Computing				
Scheme and Credits:	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 8th	3	-	2	4	

Course Objectives: The course objectives are:

- The objective of this course is to provide a clear understanding of cloud computing concept and cloud services (SaaS, PaaS, IaaS).
- To understand how business agility in an organization can be created.
- Evaluate the deployment of web services from cloud architecture.
- Compare the economic benefits delivered by various cloud models and study the best practicemodel to apply when developing and deploying cloud-based applications.

UNIT-I No of Lectures: 14

Computing Paradigms: High-Performance Computing, Parallel Computing, Distributed Computing, Cluster Computing, Grid Computing, Cloud Computing, Bio computing, Mobile Computing, Quantum Computing, Optical Computing, Nano computing.

Cloud Computing Fundamentals: Motivation for Cloud Computing, The Need for Cloud Computing, Defining Cloud Computing, Definition of Cloud computing, Cloud Computing as a Service, Cloud Computing as a Platform, Principles of Cloud computing, Five Essential Characteristics, Four Cloud Deployment Models

UNIT-II No of Lectures: 14

Cloud Computing Architecture and Management: Cloud architecture, Layer, Anatomy of the Cloud, Network Connectivity in Cloud Computing, Applications on the Cloud, Managing the Cloud, Managing the Cloud Infrastructure, Managing the Cloud applications, Migrating Application to Cloud, Phases of Cloud Migration Approaches for Cloud Migration.

UNIT-III No of Lectures: 14

Cloud Service Models: Infrastructure as a Service, Characteristics of IaaS. Suitability of IaaS, Pros and Cons of IaaS, Summary of IaaS Providers, Platform as a Service, Characteristics of PaaS, Suitability of PaaS, Pros and Cons of PaaS, Summary of PaaS Providers, Software as a Service, Characteristics of SaaS, Suitability of SaaS, Pros and Cons of SaaS, Summary of SaaS Providers, Other Cloud Service Models.

UNIT-IV No of Lectures: 14

Cloud Service Providers: EMC, EMC IT, Captiva Cloud Toolkit, Google, Cloud Platform, Cloud Storage, Google Cloud Connect, Google Cloud Print, Google App Engine, Amazon Web Services, Amazon Elastic Compute Cloud, Amazon Simple Storage Service, Amazon Simple Queue ,service, Microsoft, Windows Azure, Microsoft Assessment and Planning Toolkit, SharePoint, IBM, Cloud Models, IBM Smart Cloud, SAP Labs, SAP HANA Cloud Platform, Virtualization Services Provided by SAP, Sales force, Sales Cloud, Service Cloud: Knowledge as a Service, Rack space, VMware, Manjra soft, Aneka Platform

List of Books:

1. Ricardo Puttini, Thomas Erl, and Zaigham Mahmood, “Cloud Computing, Concepts, Technology and Architecture”.
2. M. J. Kevis, “Architecting The Cloud”, Wiley.
3. R.Yeluri, “Building the Infrastructure for Cloud Security”, Apress Open.
4. B. Sosinsky, “Cloud Computing Bible”.
5. Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance, Tim Mather, Subra Kumaraswamy, Shahed Latif, O’Reilly, SPD, rp 2011

Course Code: PEC81420	**Course Title:** Full Stack Development-II				
Scheme and Credits:	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 8th	3	0	2	4	

Course Objectives: The course objectives are:

1. Understand the JavaScript and technical concepts behind Node JS
2. Build a Web Server in Node and understand how it really works
3. Use NPM and manage node packages
4. Build a web application and API more easily using Express
5. Connect to a SQL or Mongo database in NodeJS

UNIT-I No. of Lectures: 14

Introduction: Node JS, Advantages of Node JS, Traditional Web Server Model, Node.js Process Model, process of installation on different OS.

Node JS Modules: Functions, Buffer, Module, Module Types, Core Modules, Local Modules, Module. Exports,

UNIT-II No. of Lectures: 14

Node Package Manager (NPM) : NPM, Installing Packages Locally, Adding dependency in package.json, Installing packages globally, Updating packages, Creating web server, Handling http requests, Sending requests

File System: Creating Web se Fs.readFile, Writing a File, Writing a file asynchronously, Opening a file, Deleting a file, Other IO Operations

Events: Event Emitter class, Returning event emitter, Inhering events

UNIT-III No. of Lectures: 14

Database connectivity: Connection string, Configuring, working with select command, updating records, Deleting records

MongoDB: Introduction to NOSQL, Definition of NOSQL, History of NOSQL and different NOSQL Products, Introduction to MongoDB, MongoDB vs NoSQL, MongoDB Environment, collections and documents, tools, basic commands.

UNIT-IV No. of Lectures: 14

MongoDB operations: Create Database, Drop Database, Create Collection, Drop Collection, Read Operations, Write Operations, MongoDB Data Modelling, Administration

MongoDB Security, Aggregation, Indexes, Storage, Replication

List of Books:

1. Mastering Node.js by Sandro Pasquali
2. Node.js the Right Way: Practical, Server-Side JavaScript That Scale by Jim R. Wilson
3. David Hows, “The definitive guide to MongoDB”, 2nd edition, Apress Publication.
4. Sams Teach Yourself Node.js in 24 Hours by George Ornbo.
5. Daniel Perkins, “Mongo DB, Third Edition, Create Space Independent Publishing Platform, 2016.

Course Code: PEC81520	**Course Title:** Pattern Recognition				
Scheme and Credits:	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 8th	3	-	2	4	

Course Objectives: The course objectives are:

- Explain and compare a variety of pattern classification, structural pattern recognition, and pattern classifier combination techniques.
- Summarize, analyse, and relate research in the pattern recognition area verbally and in writing.
- Apply performance evaluation methods for pattern recognition, and critique comparisons of techniques made in the research literature.
- Apply pattern recognition techniques to real-world problems such as document analysis and recognition.
- Implement simple pattern classifiers, classifier combinations, and structural pattern recognizers.

UNIT-I No. of Lectures: 14

Introduction: Introduction to pattern recognition, feature detection, classification.

Probability Theory: probability theory, conditional probability and Bayes rule. Random vectors, expectation, correlation, covariance.

Linear Algebra: Inner product, outer product, inverses, eigen values, eigen vectors, singular values, singular vectors.

Bayes Decision Theory: Minimum-error-rate classification. Classifiers, Discriminant functions, Decision surfaces. Normal density and discriminant functions. Discrete features.

UNIT-II No. of Lectures: 14

Parameter Estimation Methods: Template-based recognition, feature extraction, Training Methods, maximum likelihood and Bayesian parameter estimation. K-nearest-neighbour classification, non-parametric classification, density estimation, Parzen estimation. Unsupervised learning, clustering, vector quantization, K-means.

UNIT-III No. of Lectures: 12

Dimensionality reduction: Principal component analysis - its relationship to eigen analysis. Fisher discriminant analysis - Generalised eigen analysis. Eigen vectors/Singular vectors as dictionaries. Factor Analysis, Total variability space - a dictionary learning methods. Non-negative matrix factorisation - a dictionary learning method.

UNIT-IV No. of Lectures: 16

Mixture modelling, expectation-maximization. Hidden Markov Models, Viterbi Algorithm, Baum-Welch Algorithm. Linear Dynamical Systems, Kalman Filtering. Bayesian Networks. Decision trees, multi-layer perceptron’s. Reinforcement learning with human interaction. Genetic algorithm.

List of Books:

1. Pattern Recognition by William Gibson.
2. Pattern Recognition and Classification: An Introduction by Geoff Dougherty.
3. R.O. Duda, P.E. Hart and D.G. Stork, Pattern Classification, John Wiley, 2001.
4. S. Theodoridis and K. Koutroumbas, Pattern Recognition, 4th Ed., Academic Press, 2009.
5. C.M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.

Course Code: PEC81620	**Course Title:** Advanced Computer Architecture				
Scheme and Credits:	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 8th	3	0	2	4	

Course Objectives: The course objectives are:

- To introduce fundamental concepts relevant to the art of designing a computer system that meets functional, performance and cost goals.
- To impart knowledge about examining the qualitative and quantitative computer design trade-offs.
- To enable students to understand the basic non-classical architectures such as parallel processors, multi-core chips, pipelined and VLIW machines.

UNIT-I No. of Lectures: 12

Introduction: Review of basic computer architecture, Registers, Memory, Instruction set architecture, Instruction processing. Quantitative techniques in computer design, Balancing of Sub system Bandwidth. CISC and RISC processors.

UNIT-II No. of Lectures: 12

Pipelining: Basic concepts, instruction and arithmetic pipeline, data hazards, control hazards, and structural hazards, techniques for handling hazards, Exception handling, Pipeline optimization techniques, Reservation tables.

UNIT-III No. of Lectures: 17

Parallel and Scalable Architectures: Multiprocessors and Multi-computers ,Multiprocessor System Interconnects, Cache Coherence and Synchronization Mechanisms, Three Generations of Multi-computers, Message-Passing Mechanisms, Multi-vector and SIMD Computers, Vector Processing Principles, Multi-vector Multiprocessors, Compound Vector Processing ,SIMD Computer Organizations, Scalable, Multithreaded, and Dataflow Architectures, Latency-Hiding Techniques, Principles of Multithreading, Fine-Grain Multi-computers, Scalable and Multithreaded Architectures, Dataflow and Hybrid Architectures.

UNIT-IV No. of Lectures: 15

Software for parallel programming: Parallel Models, Languages, and Compilers, Parallel Programming Models, Parallel Languages and Compiler

Memory organisation: Hierarchical memory organization, Cache memory organization, Cache performance and optimization, Techniques for reducing cache misses; Block replacement policies, Virtual memory organization, Mapping and management techniques. I/O Sub systems, Interrupt mechanisms, I/O Processors and I/O channels.

List of Books:

1. Computer Architecture: A Quantitative Approach by John Hennessy & David Patterson, Morgan Kaufmann publication.
2. Computer Organization and Design: A Hardware/Software Interface by David Patterson and John Hennessy, Morgan Kaufmann.
3. Computer Architecture & Parallel Processing by Hwang & Briggs, Tata McGraw Hill.
4. Computer Organization & Architecture: Designing for Performance by William Stallings, Pearson

Course Code: PEC81720	**Course Title:** Natural Language Processing				
Scheme and Credits:	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 8th	3	0	2	4	

Course Objectives: The course objectives are:

1. Describe the fundamental concepts and techniques of natural language processing.
2. Distinguish among the various techniques, taking into account the assumptions, strengths, and weaknesses of each.
3. Use appropriate descriptions, visualizations, and statistics to communicate the problems and their solutions.
4. Analyze large volume text data generated from a range of real-world applications.

UNIT-I No. of Lectures: 14

Introduction- Human languages, models, ambiguity, processing paradigms; Phases in natural language processing, applications. Text representation in computers, encoding schemes. Linguistics resources- Introduction to corpus, elements in balanced corpus, TreeBank, PropBank, WordNet, VerbNet etc. Resource management with XML, Management of linguistic data with the help of GATE, NLTK.

UNIT-II No. of Lectures: 14

Regular expressions, Finite State Automata, word recognition, lexicon. Morphology, acquisition models, Finite State Transducer. N-grams, smoothing, entropy, HMM, ME, SVM, CRF. Part of Speech tagging- Stochastic POS tagging, HMM, Transformation based tagging (TBL), Handling of unknown words, named entities, multi word expressions.

UNIT-III No. of Lectures: 14

A survey on natural language grammars, lexeme, phonemes, phrases and idioms, word order, agreement, tense, aspect and mood and agreement, Context Free Grammar, spoken language syntax. Parsing- Unification, probabilistic parsing, TreeBank. Semantics-Meaning representation, semantic analysis, lexical semantics, WordNet Word Sense Disambiguation- Selectional restriction, machine learning approaches, dictionary-based approaches.

UNIT-IV No. of Lectures: 14

Discourse- Reference resolution, constraints on co-reference, algorithm for pronoun resolution, text coherence, discourse structure.

Applications of NLP- Spell-checking, Summarization Information Retrieval- Vector space model, term weighting, homonymy, polysemy, synonymy, improving user queries. Machine Translation– Overview.

List of Books:

1. Daniel Jurafsky James H., An Introduction to Natural Language Processing, Third Edition, Tata McGraw- Hill
2. James H. Martin, Computational Linguistics, and Speech Recognition
3. James A. Natural language Understanding, Pearson Education

<table>
<tr><td><b>Course Code:</b> PEC81820</td><td colspan="5"><b>Course Title:</b> Wireless and Mobile Communication</td></tr>
<tr><td><b>Scheme and Credits:</b></td><td><b>L</b></td><td><b>T</b></td><td><b>P</b></td><td><b>Credits</b></td><td rowspan="2"><b>Total marks: 100</b></td></tr>
<tr><td><b>Semester:</b> 8th</td><td>3</td><td>-</td><td>2</td><td>4</td></tr>
<tr><td colspan="6"><b>Course Objectives:</b> The course objectives are:
To introduce the concepts of wireless/mobile communication using a cellular environment. To make students aware of the various propagation methods, propagations losses, and models used in cellular communication. Various network systems and generations of wireless communications are introduced.</td></tr>
</table>

UNIT-I No. of Lectures: 14

Introduction: Wireless Networks, Wireless Transmission, Frequencies for radio transmission: Introduction to Radio wave propagation, Three Basic propagation Mechanisms: Reflection, Diffraction, Scattering**,** Multipath Propagation, SINR

Examples of Wireless Communication Systems: Paging Systems, Cordless Telephone Systems, Cellular Telephone Systems

Process of a Telephone Call, Comparison of common wireless communication systems, Trends in Cellular Radio and Personal Communication

UNIT-II No. of Lectures: 14

Frequency Reuse: Concept, Channels, Co-Channel Interference, Adjacent Channel Interference, Cell splitting, Cell sectorization

Generations of Wireless Networks: 2G, 2.5G, 3G, LTE, 5G and beyond, Brief introduction to Next-generation networks.

Data Services: GPRS, EDGE, and HSPA

UNIT-III No. of Lectures: 14

Telecommunication Systems: GSM, Mobile Services, System Architecture, GSM Channels, Components of wireless communication infrastructure: MS, BS, BTS, MSC, HLR, VLR

Handoff: Concept, Inter, and Intra-cellular Handoff, Mobile assisted handoff (MAHO), soft and hard Handoff.

Trunking and Grade of Service

UNIT-IV No. of Lectures: 14

Mobile Radio Propagation: Introduction

Large Scale Path Loss: Free Space Propagation Model

Outdoor Propagation Models: Longley Rice Model, Okumura Model

Small Scale Fading and multipath: Factors influencing small scale fading, Doppler Shift

Fast and Slow Fading

Rayleigh Fading Distribution

List of Books:

1. Jochen, Schiller. "Mobile communications." (2003). Pearson Education
2. Stallings, William. *Wireless communications & networks*. Pearson Education India, 2009.
3. Rappaport, Theodore S. "Wireless Communications--Principles and Practice, (The Book End)." (2002):
4. Yi-Bing, Imrich Chlamtac, and I. M. R. I. C. H. Chlamtac. "Wireless and Mobile Network Architecture." (2001).
5. Tse, David, and Pramod Viswanath. *Fundamentals of wireless communication*. Cambridge university press, 2005.

Course Code: PEC81920	**Course Title:** Data Analytics with R				
Scheme and Credits:	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 8th	3	0	2	4	
Course Objectives: The course objectives are: • Analyse and configure R software for statistical programming environment and describe generic programming language concepts implemented in a high-level statistical language. • Establish Program in R environment to create custom analytical models to meet the dynamic business needs. • Evaluate and verify the analysis findings by conducting various statistical tests used for hypothesis testing. • Visualize and customize the various graphical packages for creating various types of graphs, plots and charts. • Review advanced data science concepts using predictive analytics fundamentals.					

UNIT-I No. of Lectures: 14

Installation and development environment overview: downloading and installing R from CRAN, installing R on your windows computer, installation R studio, libraries in R and R studio**.**

Core Programming Principles: discover the basic data types, variables and operators in R, conditional statements, loops, functions and packages in R, using console of R Studio.

Vectors and matrices: learn how to work with vectors and matrices in R

Factors: R stores categorical data in factors, learn how to create subset and compare categorical data

UNIT-II No. of Lectures: 14

Data frames: creating, merging, naming, filtering, indexing and selection in data frames

Lists: naming, extracting, adding, deleting components from lists, sub-setting a list

Data Preparation: Gsub and sub, dealing with missing data, Handling NA values, replacing a NA values

UNIT-III No. of Lectures: 14

Data input and output in R: CSV files, excel files and SQL with R.

Advanced R programming: Mathematical functions, apply family of functions, regular expressions, dates and timestamps.

Data manipulation with R using: data filters, handling missing data, dplyr, tidyr, pipe.

UNIT-IV No. of Lectures: 14

Text mining in R: Text mining functions, string functions used in R, analysing text data for mining

Social media data mining: Facebook data analysis, twitter data analysis

Data visualization with R: Explanation and Implementation of Basic types of graphs (SCATTER PLOT, LINE CHART, BAR CHART, PIE CHART), Explanation and Implementation of Advanced types of graphs (Word Cloud, Heat Map, Bollinger Band, etc.), Dynamic Visualization using GGPLOTS, Advanced Visualization using PLOTLY, Implementation of DASHBOARDS using R MARKDOWN

List of Books:

1. R in A Nutshell By Joseph Adler, O'reilly
2. R Programming for Beginners By Sandip Rakshit, Mc Graw Hill
3. Hands on Programming With R: Write Your Own Functions and Simulations By Garrett Grolemund, O'reilly

Course Code: PEC82020	**Course Title:** Graph Theory				
Scheme and Credits:	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 8th	3	-	2	4	

Course Objectives: The course objectives are:

1. To learn the basic terminology and results concerning graphs.
2. To learn proof techniques and algorithms involving graphs.
3. To learn how to apply computer programs to study graphs.
4. To learn about open problems in graph theory.

UNIT-I No. of Lectures: 14

Scope of the course, Application areas in CS, A feel of some advanced problems in Graph Theory Introduction: Graph Terminology, Incidence and Degree, Isolated vertex, pendant vertex and Null Graph, Isomorphism, Walks, Paths and Circuits, Connected Graphs, Disconnected graphs and Components, Euler Graphs, Operations on graphs, Hamiltonian paths and circuits, The Travelling salesman problem, Konigsberg bridge problem, Three utility problem.

UNIT-II No. of Lectures: 14

Trees: Properties of Trees, Distance and Centres in a tree, Rooted and Binary Trees, Spanning Trees. Algorithms for finding minimal spanning tree: Kruskal’s algorithm, Prim’s algorithm. Cut-sets and Cut-Vertices: Cut-sets, All cut-sets in a graph, Fundamental Circuits and cut-sets, connectivity and separability, Network Flows, 1-isomorphism, 2-isomorphism.

UNIT-III No. of Lectures: 14

Planar and Dual Graphs: Planar Graphs, Kuratowski’s two graphs, Kuratowski's Theorem, Detection of planarity, Geometric dual, Combinatorial dual. Matrix Representation of Graphs: Incidence matrix, Circuit matrix, Cut-set matrix, path matrix and Adjacency matrix.

Coloring, Covering, and Partitioning: Chromatic number, Chromatic partitioning, Chromatic polynomial, Matching, Coverings, The Four Color problem.

UNIT-IV No. of Lectures: 14

Directed Graphs: Types of digraphs, Euler Digraphs, Trees with directed edges, Matrix representation of digraphs, Tournaments, Acyclic digraphs and decyclization.

Graph theoretic Algorithms: Shortest path algorithms, Dijkstra’s algorithm, Warshall - Floyd algorithm, Depth-First search in a graph, Breadth–First search in a graph.

List of Books:

1. Douglas B. West, Introduction to Graph Theory, Prentice Hall India Ltd., 2001 2.
2. Narasingh Deo, Graph theory, PHI, 1979. 3. Robin J. Wilson,
3. Introduction to Graph Theory, Longman Group Ltd., 2010.
4. R. Diestel, Graph Theory, free online edition, 2016.

OPEN ELECTIVES

<table>
<tr><td><b>Course Code:</b> OEC7120</td><td colspan="5"><b>Course Title:</b> E-Commerce and Cyber Laws</td></tr>
<tr><td><b>Scheme and Credits-</b></td><td><b>L</b></td><td><b>T</b></td><td><b>P</b></td><td><b>Credits</b></td><td rowspan="2"><b>Total marks: 100</b></td></tr>
<tr><td><b>Semester:</b> 7th</td><td>3</td><td>-</td><td>-</td><td>3</td></tr>
<tr><td colspan="6"><b>Course Objectives:</b> The objectives of the course are:
• To introduce and explore the use of information technology in all aspects of business.
• To understand various cyber risks and cyber laws.</td></tr>
</table>

UNIT-I No. of Lectures: 12

Business Models and Technological Infrastructure: Electronic Commerce and electronic business, Incentives for engaging in electronic commerce, Advantages and disadvantages, Impact of E-commerce on business. E-Commerce Models: Network Infrastructure for E-Commerce, Internet and Intranet based E-commerce, Mobile Commerce, Characteristics, Wireless Application Protocol, Mobile Information device profile.

UNIT-II No. of Lectures: 10

Web Security and Protocols: Vulnerabilities on the web. SQL injection. Cross site request forgery, Data breach, Importance and types of firewalls, Factors to consider in firewall design, Limitation of firewalls; Digital Signatures, SET protocol, SSL protocol.

UNIT-III No. of Lectures: 10

Electronic Payment Systems and Payment Gateway: Electronic Payments-Overview, E-cheques, ATM transactions, Credit cards, Debit cards, Cash cards, POS, Internet banking, Payment gateway architecture and features, Mobile banking, Mobile wallets.

UNIT-IV No. of Lectures: 14

Cyber World: Introduction – cyber space – cybercrimes – types: cyber stalking, forgery and fraud, crime related to IPR (copyright issues in cyber space, trademark issues in cyber space, software patenting issues), cyber terrorism, and computer vandalism.

Cyber Regulations- Cyber Law (an overview) scope of cyber laws (e-commerce, online contracts, IPRs, e taxation, e-governance and cybercrimes), issues relating to investigation, cyber forensic, relevant provisions under IT Act 2000, IPC and Evidence Act etc.

List of Books:

1. Ravi Kalakota, Andrew Winston, "Frontiers of Electronic Commerce", Addison-Wesley.
2. Laudon, "E-Commerce: Business, Technology, Society", Pearson Education.
3. S. Bajaj and Nag, "E-Commerce the cutting edge of Business", TMH.
4. Turban, "Electronic Commerce 2004: A Managerial Perspective", Pearson Education.

Course Code: OEC7220	**Course Title:** LaTex				
Scheme and Credits:	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester:7th	2	0	2	3	

Course Objectives: The objectives of the course are:

- Use the preamble of LaTeX file to define document class and layout options.
- Use nested list and enumerate environments within a document.
- Use tabular and array environments within LaTeX document.
- Use various methods to either create or import graphics into a LaTeX document.
- Use BibTeX to maintain bibliographic information and to generate a bibliography for a particular document.

UNIT-I No. of Lectures: 10

Introduction: Installation of the software LaTeX, Installing Extra Packages, Basics, Understanding Latex compilation.

Preparing an input file, Sentences and paragraphs, Quotation marks, Special symbols Emphasizing text -Preventing line breaks, Footnotes

UNIT-II No. of Lectures: 12

Document class, Page style, Page numbering, creating lists, Formatting lengths, Parts of the document, Title, Abstract, Dividing the document.

Common Elements: Basic Syntax, Fonts, Color, Special Characters, Writing equations, Matrix, Tables.

UNIT-III No. of Lectures: 12

Page Layout: Titles, Abstract, Chapters, Sections, Nomenclature, Customizing Page Headers and Footers, References, Equation references, citation. List making environments, Table of contents, generating new commands, Figure handling numbering, List of figures, List of tables, generating index, Importing Graphics, Floats, Figures and Captions, Footnotes and Margin Notes.

UNIT-IV No. of Lectures: 12

Packages: Geometry, hyperref, amsmath, amssymb, algorithms, algorithmic graphic, color, tilez listing, etc. Classes: Article, book, thesis, report, beamer, slides. IEEtran.

Applications: Writing Resume, writing question paper, writing articles/research papers, Presentation using beamer.

List of Books:

1. Kottwitz, S., LaTeX Beginner's Guide, Packt Publishing, 2011.
2. Kopka, H., Daly, P.W., Guide to LaTeX, Addison-Wesley Professional; 4th edition, 2003.
3. Grätzer, G., More Math Into LaTeX, Springer, Revised 5th edition, 2016.

Course Code: OEC8120	**Course Title:** Entrepreneurship Development and Management				
Scheme and Credits:	**L**	**T**	**P**	**Credits**	**Total marks: 100**
Semester: 8th	3	-	-	3	
Course Objectives: The objectives of this course are to explain the key ingredients to be a successful entrepreneur appreciate the value of entrepreneurship in daily life especially in marketing and understand the impact of social and environmental issues in starting up of a new venture. To develop entrepreneurial culture and provide opportunity to build analytical and practical skills.					

UNIT-I No. of Lectures: 12

Entrepreneurship Development: Meaning, Objectives, Type of Entrepreneurs, Importance of Entrepreneurship Training, Factors affecting Entrepreneurship, Linkage between Entrepreneurship and Economic Development.

Entrepreneurship Support System: Small Industries Development Bank of India, District Industrial Centres and other Supporting Agencies, types of start-ups.

UNIT-II No. of Lectures: 14

Organizational Behaviour and HRM: Evolution of organizational behaviour, personality-personality traits-MBTI-Big-five model, leadership- theories of leadership-trait theory-path-goal theory.

Recruitment and Selection: Motivation-Herzberg's-Maslow's theories, Organization structure-line-line and functional structures, training and development.

UNIT-III No. of Lectures: 12

Marketing Management: Concepts of marketing, marketing mix-product-price-place-promotion, product life cycle, new product development, pricing-objectives and strategies, Segmentation, Targeting and Positioning

Product and Finance Management: Plant location-factors affecting plant location, plant layout-process- product layout, inventory control, EOQ. Wealth maximization and profit maximization.

UNIT-IV No. of Lectures: 10

Project Management: Project Report Preparation: Identifying Business Opportunities, Project Report and its Importance,

Quantitative Tools for Entrepreneurs: Construction of network projects, PERT and CPM, calculation of earliest start, earliest finish, latest start and latest finish of activities crashing of activities (Basic concepts only).

List of Books:

1. Ram Chandran, 'Entrepreneurial Development', Tata McGraw Hill, New Delhi
2. Badhai, B 'Entrepreneurship for Engineers', Dhanpat Rai & co. (p) Ltd.
3. Desai, Vasant, ' Project Management and Entrepreneurship', Himalayan Publishing House, Mumbai, 2002.
4. Ramaswamy V S and NamaKumari : Marketing Management, McMillan India Ltd
5. Philip Kotler and Abraham Koshy : Principles of Marketing, Pearson Education, New Delhi
6. McShane, S.L. and Von Glinow, M.A., Organizational Behaviour, New Delhi, Tata McGrawHill Publishing company ltd.
7. P. Jyothi, P. and Venkatesh, D.N., Human Resource Management, New Delhi, Oxford University Press.

Course Code: OEC8220	**Course Title:** Multimedia Technology				
Scheme and Credits:	**L**	**T**	**P**	**Credits**	**Total Marks: 100**
Semester: 8th	3	0	0	3	

Course Objectives: The objectives of the course are:

- Understanding firm grounding in the fundamentals of the underpinning technologies in graphics, distributed systems and multimedia
- Understanding principled design of effective media for entertainment, communication, training and education
- Familiarisation with Virtual reality system.

UNIT-I No. of Lectures: 10

Concept of Non-Temporal and Temporal Media. Basic Characteristics of Non-Temporal Media; Images, Graphics, Text. Basic Characteristics of Temporal Media: Video, Audio, and Animation. Hypertext and Hypermedia. Presentations: Synchronization, Events, Scripts and Interactivity, Introduction to Authoring Systems.

UNIT-II No. of Lectures: 12

Compression techniques, Basic concepts of Compression. Still Image Compression: JPEG Compression. Features of JPEG2000. Video Compression: MPEG- 1&2 Compression Schemes, MPEG-4 Natural Video Compression. Audio Compression: Introduction to speech and Audio Compression, MP3 Compression Scheme.

UNIT-III No. of Lectures: 12

Multimedia systems architecture, General Purpose Architecture for Multimedia Support: Introduction to Multimedia PC/Workstation Architecture, Characteristics of MMX instruction set, I/O systems: Overview of USB port and IEEE 1394 interface, Operating System Support for Multimedia. Multimedia Database Design, Content Based Information Retrieval: Image Retrieval, Video Retrieval, Overview of MPEG-7, Design of video-on-Demand Systems.

UNIT-IV No. of Lectures: 10

Introduction to Virtual Reality and Virtual Reality Systems, Related Technologies
Teleoperation and Augmented Reality Systems Interface to the Virtual World-Input; Head and hand trackers, data globes, hap tic input devices. Interface to the Virtual World- Output, Stereo display, head mounted display, auto-stereoscopic displays, holographic displays, hap tic and force feedback.

List of Books:

1. Multimedia System Design, Andleigh and Thakarar , PHI
2. Multimedia Technology & Application, David Hillman, Galgotia Publications.
3. Multimedia Computing Communication and Application, Steinmetz, Pearson Education.
4. Virtual Reality Systems, John Vince, Pearson Education.
5. Fundamentals of Computer Graphics and Multimedia, D.P. Mukherjee, PHI