PARVATIBAI CHOWGULE COLLEGE OF ARTS AND SCIENCE(AUTONOMOUS)
DEPARTMENT OF COMPUTER SCIENCE

COURSE STRUCTURE

THREE YEAR B.Sc. DEGREE COURSE IN COMPUTER SCIENCE

To be offered from: (2022 -2023)

Programme Specific Outcome (PSO) for B.Sc. Computer Science

At the end of the programme the students will be able to:

PSO1: Analyse a Software problem, design, implement a solution and evaluate the proposed solution to ensure that it meets customer needs and Software standard.

PSO2: Use and Apply appropriate current technologies, techniques and modern tools necessary for computing practice.

PSO3: Embark on an Entrepreneurial venture or be eligible for employment in IT industry or pursue higher education.

PSO4: Apply the concept of networking and security features in designing the systems.

APPROVED COURSE SYLLABI FOR SEMESTER III, IV, V AND VI

PARVATIBAI CHOWGULE COLLEGE OF ARTS AND SCIENCE(AUTONOMOUS)

DEPARTMENT OF COMPUTER SCIENCE

COURSE STRUCTURE

THREE YEAR B.Sc. DEGREE COURSE IN COMPUTER SCIENCE

(2022 -2023)

SEMESTER	CORE COMPULSORY		CORE ELECTIVE			
I	COM-I.C-1 Mathematical foundation of Computer Science–I	COM-I.C-2 *Introduction to Programming	- - -	---	---	---
II	COM-II.C-3A ** Database Management System	COM-II.C-4* Data Structures	- - -	---	---	---
III	COM-III.C-5A* Object Oriented Programming	---	COM-E1 Software Engineering	COM-E2 Digital Logic Design	COM-E3 Mathematical Foundation of Computer Science–II	COM-E4 Web Designing

IV	COM-IV.C-6 Computer Architecture and Organization	- - -	COM-E5 Design & Analysis of Algorithms	COM-E10 Mobile Application Development	COM-E7 Server Side Programming	COM -E8 HCI
V	COM-V.C-7* Operating Systems	- - -	COM-E9 Embedded Systems	COM-E17 Data Base Management System II	COM-VI-E 18 Web Development Framework	COM-E12 Software Testing
VI	COM-VI.C-8* Computer Networks	- - -	COM-E-13 Network Security	COM-E-14 Cloud Computing	COM-.E-15 Introduction to Data Science	COM E-19 Web Development with Laravel

Note: *Core Compulsory Courses also offered form in or subject combination.

** Core Compulsory Courses also offered for minor subject combination in 4th Semester.

SEMESTER III

Course Title: Object Oriented Programming
Course Code: COM-III.C-5A
Marks: 75
Credits: 3
Duration: 45 Hrs

Course Prerequisites: Nil

Course Objectives:

- To teach the basic concepts and techniques which form the object oriented programming paradigm.
- To introduce object oriented programming (OOP) using Java.

Course Outcomes:
Upon completion of the course students will be able to:

CO1: Apply fundamental object-oriented concepts in problem solving.
CO2: Analyse problem scenario and identify classes/objects, their properties/functionalities and associations.
CO3: Analyse the problem scenario and model the system using UML diagrams.
CO4: Implement the object oriented model in any object oriented language.

SYLLABUS:

UNIT I: Introduction **[15 HRS]**
Principles of OOP

Programming paradigms. Basic concepts in OOP. OOP: major principles - encapsulation, abstraction, inheritance, polymorphism. Benefits of OOP. Applications of OOP.

Introduction to Java
Basics of Java programming, Data types, Variables, Operators, Control structures including selection, Looping, Java methods, Overloading, java. Math class, Arrays in java.

UNIT II Principles of OOP **[15 HRS]**
Basics of objects and classes in java, Constructors, Finalizer, Visibility modifiers, Methods and objects, object serialization, Inbuilt classes like String, Character, String Buffer, File, this reference.

Inheritance and Polymorphism:
Inheritance in java, Super and sub class, Overriding, java.lang. Object class, Polymorphism, Dynamic binding, Generic programming, Casting objects, Instance of operator, Abstract class, Interface in java, Package in java, java.util package.

UNIT III: Exceptions, Multi-threading and GUI programming: **[15 HRS]**
Event and GUI programming
Design patterns – what and why? It's classification. Introduce the Observer design pattern.

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, Applet and its life cycle.

Multi-threading in java
Multi-threading in java, Thread life cycle and methods, Runnable interface, Thread synchronization.

Exception handling
Exception handling – what and why? Try and catch block. Multiple catch blocks. Nested try, finally block, throw keyword, throws keyword. Custom Exception. Introduction to the Collections Framework.

REFERENCES:

MANDATORY:
1.Deitel, P., & Deitel, H. (2011). Java How to program. Prentice Hall Press.

SUPPLEMENTARY:

1. Naughton.P, Schildt.H, (2000), Java 2 – The Complete Reference TMH publications
2. Patrick,N. (1997). The Java Handbook –TMH publications
3. Mughal, K. A., & Rasmussen, R. W. (2003). A programmer's guide to Java certification: a comprehensive primer. Addison-Wesley Professional.
4. Flanagan, D. (2004). Java examples in a nutshell.
5. Arnold, G., H.(2005)“The Java Programming Language” Addison-Wesley Professional.

WEB BASED:
1.www.javapoint.com
2.www.tutorialspoint.com
3.www.gurugg.com
docs.logout.org > Programmation >Java>Programming with Java_A primer

Practicals: Object Oriented Programming
Credit: 1
Marks: 25
Duration: 30 Hrs

Programs using Java language that covers the following concepts:
1) Classes and instances
2) Working with the java.Math class
3) Inheritance
4) Composition v/s inheritance
5) Polymorphism, abstract classes and interfaces
6) Algorithm and Data Structures
7) Utilizing the java.util package
8) Event handling and GUI

9) Applets
10) I/O programming
11) Exception handling
12) Multi-threading
13) Collections framework

Course Title: Software Engineering
Course Code: COM-E1
Marks: 75
Credits: 3
Duration: 45 Hrs

Course Prerequisites: Nil

Course Objectives:
- To understand the various software development methodologies and estimation of software projects.
- To analyse and design software projects.
- To study the various phases of a S/W Development Project.

Course Outcomes:
Upon completion of the course students will be able to:

CO1: Understand the various Software Development Methodologies.
CO2: Apply Estimation techniques to live projects.
CO3: Analyse Software Projects.
CO4: Design Software Projects.

SYLLABUS:

UNIT I: **[15 HRS]**

SOFTWARE PROCESS:
Characteristics of software process.
Software Development Processes and Methodologies: waterfall, prototyping, iterative, spiral, unified process. Benefits of iterative and incremental approach with emphasis on Unified process, CASE Tools, Agile methodologies.

PROJECT MANAGEMENT:
Planning a Software Project – Cost estimation, Project Scheduling, Software configuration management plans, Quality Assurance plans, Project Monitoring plans and Risk Management.
Techniques such as Interviewing, Requirement Workshop, brainstorming, prototyping. Characteristics of SRS.

UNIT II: **[15 HRS]**

OOAD and UML:
OOAD: Definition; object oriented analysis; object oriented design and modelling; Assigning responsibilities.

UML: Main UML diagrams – Class diagram, sequence diagram, activity diagram, use case diagram.
Use case model – use case diagram, use case descriptions, use case realization using sequence and activity diagrams. Supplementary requirements. Advanced use case model features.

Requirements: Functional and non-functional
System Design : Class diagram, sequence diagram, activity diagram, state chart diagram, deployment diagram. Brief introduction to other UML diagrams.

SOFTWARE ARCHITECTURE PATTERNS:
Major Architectural Styles (patterns) like Layered Architecture, Pipe and Filter, Shared (Central)Data Store, Event Driven, Model-View-Controller (MVC), "Distributed & Emerging" Service Oriented Architecture (SOA) and Elementary GRASP Patterns.

UNIT III **[15 HRS]**

HUMAN COMPUTER INTERACTION:
HCI Definition; User categories, Interface Design-Internal & External Interface design, user interface design, Interface design guidelines.

CODING:
Coding styles, standards, peer reviews, checklist.

TESTING:
Testing Fundamental, Functional Testing, Structural Testing, Testing Object-Oriented Programs, Testing Process and Metrics.

DOCUMENTATION and MAINTENANCE:
Need for Software Documentation. Types of documentation, Need for Maintenance; Types of Maintenance.

REENGINEERING:
Business Process Reengineering, Software Reengineering, Reverse Engineering, Restructuring, Forward Engineering, The Economics of Reengineering.

REFERENCES:

MANDATORY:

1. Pressman, R. S. (2005). Software engineering: a practitioner's approach. Palgrave Macmillan.
2. Larman C.,(2015) Applying UML and patterns. Addison Wesley.
3. Bourque, P., & Fairley, R. E. (2014). Guide to the software engineering body of knowledge (SWEBOK (R)): Version 3.0. IEEE Computer Society Press.

SUPPLEMENTARY:

1.Jalote, P. (2012). An integrated approach to software engineering. Springer Science & Business Media.
2. Sommerville I.,(2015) Software Engineering .Adison Wesley.
3. Fowler, M. (2003). UML Distilled: A Brief Guide to the Standard Modelling Object Language. Object Technology Series, 3rd edition, Addison-Wesley.

WEB BASED:
1.https://www.tutorialspoint.com/software_engineering
2.https://www.w3schools.in/sdlc-tutorial
3.https://www.geeksforgeeks.org/software-engineering
4.https://www.javatpoint.com/software-engineering-tutorial

Practicals : Software Engineering

Credit : 1
Marks : 25
Duration: 30 Hrs

List of suggested PRACTICALS :
For a given project/case study
1) Requirements Gathering Techniques [2P]
2) Gantt Chart [2P]
3) USE Case diagram and Use Case descriptions for the Use Cases [3P]
4) Class Diagram [2P]
5) Sequence Diagram [2P]
6) Activity Diagram [2P]
7) State Chart Diagram [2P]

Course Title: Digital Logic Design
Course Code: COM - E2
Marks: 75
Credits: 3
Duration: 45 Hours

Course Prerequisites: Nil

Course Objectives:

- To understand the basics of Boolean algebra and the operation of logic components, combinational, sequential circuits and design of digital systems.

Course Outcomes:
Upon completion of the course students will be able to:

CO1: Convert values between various number systems/codes.
CO2: Simplify the logical expression using Boolean algebra.
CO3: Design, simplify and implement combinational logic circuits.
CO4: Design and implement the sequential logic circuit and their applications.
CO5: Explain the methods of D/A converter and A/D converters (Successive Approximation method).
CO6: Explain the classifications and characteristics of semi conductor memories with memory expansion.

SYLLABUS:

UNIT I:Number Systems, Boolean Algebra and Convertors: [15 HRS]
Number Systems: Decimal, Binary, Hexadecimal, and Octal number systems; BCD Code, Gray Code, Excess-3 Code, ASCII code, Unicode.
Boolean Algebra: Basic Boolean functions, Postulates and theorems of Boolean Algebra, logic gates, Sum-of-Products and Product-of-Sums forms of Boolean functions; Canonical and Standard forms. Simplification of Boolean Functions, Plotting of K-Maps, POS and SOP Simplification, NAND and NOR implementation.
D/A & A/D Converters: Digital to Analog converter – Introduction, Weighted-Resistor & R-2R Ladder , Analog to Digital Converter – Introduction, Successive Approximation method.

UNIT II: Combinational and Sequential Circuits: [15 HRS]
Combinational Circuits: Design procedure for combinational logic circuits; design and analysis of Half Adder, Full Adder; their use in designing other combinational logic circuits; Analysis & Design of Encoders and Decoders; Multiplexer and demultiplexers; their use in designing combinational circuits.
Sequential Circuits (Registers):SR, JK, T, D Flip-flops and Latches, their schematic symbols, Truth tables. Shift Registers - SISO, SIPO, PISO, PIPO, Bi-directional Shift Registers, Loading methods for Shift, Registers.

UNIT III: Sequential Circuits(Counters) and Semiconductor memories: [15 HRS]
Counters: Design and analysis of Counters: Synchronous Counters, Modulo Counters ,Asynchronous, Ripple and Ring Counters; Application of Counters.

Semiconductor memories: Memory organization and operation, expanding memory size, Classification and characteristics of memories, RAM, ROM, Synchronous DRAM SDRAM), Double Data Rate SDRAM, Synchronous SRAM, DDR and QDR SRAM ,Content Addressable Memory.

REFERENCES:

MANDATORY:

1. Jain, R. P. (2003). Modern digital electronics. Tata McGraw-Hill Education.

SUPPLEMENTARY:

1. Leach.D, Malvino, M , Saha.G, Digital Principles and Applications, ,Mc. Graw Hill (SiE).
2. Mano, M. M. (2017). Digital logic and computer design. Pearson Education India.
3. Taub, H., & Schilling, D. L. (1977). Digital integrated electronics. McGraw-Hill College.

WEB BASED:

1. https://nptel.ac.in/courses/117/106/117106114/
2. https://www.geeksforgeeks.org/digital-electronics-logic-design-tutorials/
3. https://logisim.en.uptodown.com/windows
4.https://www.edutechlearners.com/download/books/Morris%20Mano%20Digital%20Design%203rd%20Edition.pdf
5.https://books.google.co.in/books?id=dnq3HmDN1ZAC&printsec=frontcover&redir_esc=y#v=one page &q & amp;f=false

Practicals: Digital Logic Design
Credits: 1
Marks: 25
Duration: 30 Hrs

Practicals:

1. Introduction to digital ICs, specifications, verification of the truth tables of logic gates1 (2 Lab sessions)
2. Introduction to digital ICs, specifications, verification of the truth tables of logic gates2. (2 Lab sessions)
3. Implementation of the given Boolean function using logic gates in SOP form.(adder/subtractor -2 sessions)
4. Decoder and Encoder (2 sessions).
5. Multiplexer –Demultiplexer (2 sessions).
6. Verification of state tables of RS, JK, T and D flip-flops.
7. Design and verification of the 4-bit asynchronous and Synchronous counter (3 sessions)

Course Title: Mathematical Foundation of Computer Science - II
Course Code: COM - E3
Credits: 3
Marks: 75
Duration: 45 Hrs

Course Prerequisites: Nil

Course Objectives:
- To build mathematical foundations in the areas namely graph theory.
- Application of numerical analysis related to topics of computer science.

Course Outcomes:
On completion of the course students should be able to:

CO1: System of linear equations in solving the problems
CO2: Apply the Interpolation methods for solving the problems numerically.
CO3: Demonstrate the process of curve fitting of data.
CO4: Determine the roots of polynomial equations.

SYLLABUS:

UNIT I: Systems of Linear Equations and Matrices, Linear Combinations and Linear Independence **[10 HRS]**
Systems of Linear Equations, Matrices and Elementary Row Operations, The Inverse of a Square Matrix, Matrix Equations, Applications of Systems of Linear Equations. Linear Combinations and Linear Independence.

UNIT II: Vector Spaces, Linear Transformations, Eigenvalues and Eigenvectors [15 HRS]
Definition of a Vector Space, Subspaces, Basis and Dimension, Coordinates and Change of Basis Linear Transformations, The Null Space and Range, Isomorphisms, Matrix Representation of Linear Transformations, Similarity. Eigen values and Eigen vectors, Diagonalization.

UNIT III: Interpolation, Numerical Integration, algebraic and transcendental equations[20 HRS]
Introduction; Various methods of interpolation; Various methods of curve fitting; Newton's method of forward interpolation formula; Newton's method of backward interpolation formula. Lagrange's formula.
General quadrature formula; Trapezoidal rule; Simpson's one-third rule; Simpson's three-eighth rule; Weddle's rule.
Graphical method; Bisection method; Method of false position; Secant method; Newton-Raphson method. Adjoint, inverse of a matrix; Rank; Linear equations; Characteristics roots and vectors.

REFERENCES:

MANDATORY:

1. Defranza, J., & Gagliardi, D. (2015). Introduction to Linear Algebra with applications. Waveland Press.
2. Parthasarathy.K,(1994) Basic Graph Theory, Tata McGraw-Hill Publishing
3. Goel,B., & Mittal,S. (1998) Numerical Analysis, Pragati Prakashan,
4. Iyengar,S.N.,(2010) Matrices, Anmol Publications.

SUPPLEMENTARY:

1. Clark, J., & Holton, D. A. (1991). A first look at graph theory. World Scientific.
2. Chatterjee,P. (1996) Numerical Analysis, RajhansPrakashanMandir.
3. Krishnamurthy,V. (1976) Introduction to Linear Algebra, Affiliated East-West Press.

WEB BASED:

1.https://www.cse.iitb.ac.in

Practicals: Mathematical Foundation of Computer Science - II
Credit: 1
Marks: 25
Duration: 30 Hrs

Linear equations
1. Systems of Linear Equations
2. Linear Transformations
3. Matrix Representation of Linear Transformations

Numerical Analysis
4) Find the value of dependent variable using Newton's forward formula for a given value of independent variable.
5) Use Newton's backward formula to estimate a value.
6) Estimate a value using Lagrange's formula.
7) Apply Simpson's three-eighth rule to find the value of integration.
8)Apply Newton-Raphson method OR secant method to estimate the root of aequationLinear Algebra.
9) Find the rank of a matrix.
10) Find solutions of a system of equations
11) Find the Eigen values and Eigen vectors

Course Title: Web Designing
Course Code: COM-E4
Marks: 75
Credits: 3
Duration:45 Hrs

Course Prerequisites: Nil

Course objectives:

- Design good user interfaces.
- Apply design principles such as learnability, visibility, error prevention, efficiency and graphic design.

Course Outcomes:
On completion of the course students will be able to:

CO1: Design Content for a web application.
CO2: Style content so as to provide an effective User Interface.
CO3: Provide for dynamism in the User Interface to enhance usability.
CO4: Develop a static web application.

SYLLABUS:

UNIT I: Structuring the UI with HTML/HTML5 [15 HRS]
User Interface – Introduction, its importance, design principles – learnability, visibility, error prevention, efficiency, graphic design. Design Patterns for GUI – View tree, Listener, Widget, Model-View Controller.

HTML - Introduction. The development process, basic HTML, formatting and fonts, commenting code, colour, hyperlink, lists, tables, images, simple HTML forms, web site structure, Meta tags, Character entities, frames and frame sets.

HTML5 - Introduction, New Elements, Canvas, SVG, Drag/Drop, Geolocation, Video, Audio, Input types, form elements, form attributes, semantic, web storage, app cache, web workers, SSE

UNIT II : Styling the UI with CSS/CSS3 [10 HRS]
CSS - Introduction – Syntax, Id & Class, Backgrounds, Text, Fonts, Links, Lists, Tables. CSS Box Model – Border, Outline, Margin, Padding. Advanced - Grouping/Nesting, Dimension, Display, Positioning, Floating, Align, Pseudo-class, Pseudo-element, Navigation Bar, Image Gallery, Image Opacity, Image Sprites, Media Types, Attribute Selectors.
CSS3 - Introduction, Borders, Backgrounds, Gradients, Text Effects, Fonts, 2D Transforms, 3D Transforms, Transitions, Animations, Multiple Columns.

UNIT III: Dynamism in UI [20 HRS]
JavaScript - Introduction - What is JavaScript, Understanding Events, JavaScript Example, External JavaScript. Basic Elements – Comment, Variable, Global Variable, Data Types, Operators, If Statement, Switch, Loop: for and while, Function. JavaScript Objects – objects, Array. Browser Object Model - Browser Objects, Window Object, Document Object – get Element ById, get Elements ByName,

getElementsByTagName, innerHTML property, inner Text property. Validation- form validation, email validation.

JQuery : Introduction - Syntax, Selectors, Events. Effects- Hide/Show, Fade, Slide, Animate, stop(), Call back, Chaining. HTML/CSS- Add, Remove, CSS Classes, css(), Dimensions, slider. Traversing – ancestors, descendants, siblings, filtering.

REFERENCES:

MANDATORY:

1. D.T,(2018) Web Technologies, Black Book, Dream Tech

WEB BASED:

1.https://www.w3schools.com
2.https://www.tutorialspoint.com/html/index.htm
3.https://www.tutorialspoint.com/css/index.htm
4.https://www.tutorialspoint.com/javascript/index.htm
5.https://www.tutorialspoint.com/jquery/index.htm
6.https://www.udemy.com/courses/development/web-development/

Practicals : Web Designing
Marks: 25
Credits: 1
Duration: 30 Hrs

List of Assignments: (the numbers in brackets indicate number of practicals) :

1) Case studies to review UI designs [2 P]

2) Create a HTML page with the following: [3 P]

a) title heading paragraph emphasis strong and image elements
b) complex HTML table
c) simple HTML Form covering major form elements
d) Embed Video in an HTML page

3) Using CSS do the following: [3 P]

a) Create a Navigation bar (with dropdown) with CSS
b) Create a CSS Grid
c) Create a CSS3 based button
d) Make an image rounded shape
e) Create a CSS based sticky footer
f) Create CSS3 Corner Ribbon
g) Create CSS3 blurry text effect
h) Create CSS3 speech bubble shape
i) Create image cross fade with CSS3 transition
j) Set style for link hover active and visited states of hyperlink

4) Write JavaScript functions to : [4 P]

a) accept a string as a parameter and converts the first letter of each word of the string in upper case.
b) check whether a given credit card number is valid or not.
c) check whether a given value is an valid url or not.
d) check whether a given email address is valid or not.
e) print an integer with commas as thousands separators
f) remove items from a dropdown list.

5) Use J Query to : [3 P]

a) Disable buttons
b) Make textbox read only
c) Uncheck check boxes
d) Confirm again
e) Sort
f) Switch rows and columns

A mini project combining all the technologies learnt using a front-end development framework such as bootstrap is recommended.

SEMESTER IV

Course Title: Computer Architecture and Organization
Course Code: Com-IV. C-6
Marks: 75
Credits: 3
Duration: 45Hours

Course Prerequisites: Nil

Course Objectives:
- Conceptualize the basics of organizational and architectural issues of a digital computer.
- Analyse processor performance improvement using instruction level parallelism.
- Explain different types of addressing modes and memory organization.
- Study various data transfer techniques in digital computer.
- Learn microprocessor architecture and study assembly language programming.
- Understand the taxonomy of Parallel Processor.
- To have a thorough understanding of the basic structure and operation of a digital computer.

Course Outcomes:
Upon completion of the course students will be able to:

CO1: Identify various components of the Computer System.
CO2: Explain the detailed function of a typical microprocessor and its control unit.
CO3: Implement Assembly Language Program for 8086 processor for a given task.
CO4: Differentiate the function and role of semiconductor memories and map the cache memory for the given scenario.
CO5: Appraise the importance of input/output modules and Interrupts and their functions.
CO6: Distinguish the characteristics and function of I/O interfaces to computer system.
CO7: Illustrate the function of pipelined architecture and classify the Multiprocessor systems.

SYLLABUS

UNIT I: Computer System and Processor Unit: [15 HRS]
Function and structure of a computer, Interconnection of components, Performance of a computer. Computer Architecture – Princeton (Von Neumann) and Harvard architecture. Architecture of 8086 processor - Registers, ALU and Control unit, Data path in a CPU. Instruction cycle, Organization of a control unit – Block Diagram of Hardwired and Micro programmed control unit.

UNIT II: Instruction Set and Memory Subsystem: [15 HRS]
Machine instructions characteristics, Types of operations-data transfer, arithmetic, logical, conversion, I/O, system control, transfer of control; 8086 Instruction Set and Assembly language: Addressing modes-immediate, direct, indirect, register, register indirect, displacement, stack. Instruction formats - instruction length, allocation of bits, variable length instructions, Instruction set architectures – CISC and RISC architectures. Characteristics of memory system, the memory hierarchy, Semiconductor memories, Types of ROM & RAM, Cache memory unit - Concept of cache memory, Organization of a cache memory unit, Mapping methods, replacement algorithms, write policy, block size.

UNIT III: Input/Output Subsystem and Parallel Processing: [15 HRS]
General block diagram of External device & I/O module, Programmed I/O, Interrupt driven I/O, DMA, I/O channels and I/O processors. I/O interfaces – Serial port, Parallel port, PCI bus, SCSI bus, USB bus, Firewire and Infiniband .Classifications, Introduction to pipeline processing: Instruction pipeline & Arithmetic pipeline, Introduction to Array & Vector processors, Introduction to Multiprocessors.

REFERENCES:

MANDATORY:
1. Stallings, W. (2003). Computer organization and architecture: designing for performance. Pearson Education India..

SUPPLEMENTARY:
1. Mano, M. M. (2005). Computer system architecture. Dorling Kindersley Pearson.
2. Patterson, D. A., & Hennessy, J. L. (2013). Computer organization and design MIPS edition: the hardware/software interface. Newness.
3. Douglas V. Hall, (2012), Microprocessors and its Interfacing, McGraw Hill Education (India) Private Limited.

WEB BASED:

1. http://williamstallings.com/ComputerOrganization/
2. https://www.tutorialspoint.com/computer_fundamentals/index.htm
3. http://www.ecs.umass.edu/ece/koren/architecture/
4. http://www.cs.colby.edu/djskrien/CPUSim/
5. https://teachcomputerscience.com/little-man-computer/
6. https://vivaxsolutions.com/web/lmc.aspx

Practicals: Computer Architecture and Organization
Credits: 1
Marks: 25
Duration: 30 Hrs

1. Study of Motherboard, Peripherals and the Computer System: O.S. Installation (Dual Boot): BIOS; Manage disk partitions: understand MBR-style partitions, (primary, extended, logical); list/create/delete partitions; Manage logical volumes: create/remove physical volumes, create/delete logical volumes, Boot loader. Installation of drivers; updating software packages
2. DOS Commands, Tools for Computer Management (Disk Management, Disk Cleanup, Defragmentation, Performance Monitor, System Restore etc).
Assembly language programs for 8086 using MASM / compatible assembler or Simulator, either in Windows or Linux.
3. Study of addressing modes.
4. Programs for arithmetic operations1
5. Programs for arithmetic operations2
6. Programs for arithmetic operations3
7. Programs for data transfer operations
8. Programs for logical operations1

9. Programs for logical operations2
10. Programs code conversion1
11. Programs code conversion2
12. Programs on sorting
13. Programs on searching
14. DOS/BIOS – Programming1
15. DOS/BIOS – Programming2

Course Title: Design and Analysis of Algorithms
Course Code : COM-E5
Marks :75
Credits :3
Duration :45 Hrs

Course Prerequisites: Nil

Course Objectives:

- To study paradigms and approaches used to analyse and design algorithms and to appreciate the impact of algorithm design in practice.
- To ensure that students understand how the worst-case time complexity of an algorithm is defined, how asymptotic notation is used to provide a rough classification of algorithms and compare with one another, and how there are still some problems for which it is unknown whether there exist an efficient algorithm, and how to design efficient algorithms.

Course Outcomes:
Upon completion of the course students will be able to:

CO1: To analyse the performance of algorithms.
CO2: Choose appropriate algorithm design techniques for solving problems.
CO3: Analyse empirical results to get a deeper understanding of the algorithmic solutions.
CO4: Apply important algorithmic design paradigms and methods of analysis.

SYLLABUS:

UNIT I: Algorithm Analysis and Divide and Conquer Strategy [15 HRS]
What is an Algorithm?, Rules for writing Algorithms, Properties of Algorithms, Framework for design and analysis of algorithms(RAM model of computation),Recursive Algorithms, Space and Time Complexity by Tabular method(Performance Analysis) Elements of Divide and Conquer Algorithms, Quick Sort algorithm, Merge sort analysis, Strassen's algorithm for matrix multiplication, Analysis of Binary Search ,The Maximum sub array Problem.

UNIT II: Dynamic programming and Greedy Techniques [15 HRS]
General Method, caching v/s computation, Fibonacci numbers by recursion, Fibonacci numbers by cashing, Fibonacci numbers by dynamic programming, Optimal Binary Search Tree, Rod Cutting Problem. Elements of greedy strategy, Activity-selection problem, Job sequencing with deadlines .Knapsack problem.

UNIT III: Graphs and Complexity Classes [15 HRS]
Elementary graph algorithms- Minimum spanning tree, growing a spanning tree, Kruskal and Prim algorithms, Breadth First search and Depth First Search, Travelling salesman problem. Introduction to polynomial time algorithms, NP, NP Complete, NP Hard, Approximation and Randomisation algorithms.

REFERENCES:

MANDATORY:

1.Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2009). Introduction to algorithms. MIT press.

SUPPLEMENTARY:

1.Horowitz.T, Sahni.S, Rajasekaran.S,(2010) "Fundamentals of Computer Algorithms", Universities Press Maryland,
2. A. Aho, J. Hopcroft, J. Ullman (2016)."The Design and Analysis of Computer Algorithms" Pearson Education India New Delhi,

WEB BASED:

1.http://www.iitk.ac.in/esc101/2009Jan/lecturenotes/timecomplexity
2. https://home.cse.ust.hk/~dekai/271/notes/L12/L12.pdf
3. https://nptel.ac.in/courses/106106131/
4. https://www.cs.cornell.edu/courses/cs3110/2012sp/lectures/lec20-master/lec20.html
5. https://www.cs.indiana.edu/~achauhan/Teaching/B401/LectureNotes/complexityClasses.html

Practicals: Design and Analysis of Algorithms
Credit: 1
Marks: 25
Duration: 30 Hrs

1 Program to find GCD of 2 numbers using Iterative approach and Recursive approach.
2 Program for quick Sort.
3 Program for Merge sort.
4 Program to perform Binary Search using Recursive approach.
5 Program to implement maximum subarray problem.
6 Program to generate Fibonacci numbers using Dynamic Programming approach.
7 Program to implement Activity Selection Problem.
8 Program to implement job sequencing with Deadlines.
9 Program to implement Knapsack Problem.
10 Program to implement Rod Cutting Problem.
11 Program to implement Binary Tree.
12 Program to represent graph using matrix/ Linked List.
13 Program to implement BFS/DFS Traversal on graph.
14 Program to implement Kruskal's Algorithm.
15 Program to implement Prim's Algorithm.

Course Title: Mobile Application Development
Course Code: COM-E10
Marks: 75
Credits: 3
Duration:45 Hrs

Course Prerequisites: Nil

Course Objective:

- Students learn how to develop applications for mobile devices, including smart phones and tablets. Students are also introduced to the current mobile platforms, mobile application development environments and mobile device input methods. Students will design and build a variety of apps on a popular platform throughout the course to reinforce learning and to develop real competency.

Course Outcome:
Upon completion of the course students will be able to:

CO1: Explain mobile devices, including their capabilities and limitations.
CO2: Review current mobile platforms and their architectures.
CO3: Develop mobile applications on a popular mobile platform.
CO4: Evaluate development with another mobile platform.

SYLLABUS:

UNIT I: Introduction to mobile devices: [15 HRS]
Mobile devices vs. desktop devices, Why we Need Mobile App, Different Kinds of Mobile Apps, ARM and intel architectures, Power Management, Screen resolution, Touch interfaces, Application deployment - App Store, Google Play, Windows Store, Development environments – Android Studio, Phone GAP, Native vs. web app, Browser-detection, Touch interfaces, Geolocation, Screen orientation, Mobile browser? interpretations? (Chrome/IE).Comparing and Contrasting architectures of Android, iOS and Windows, Underlying OS(Darwin vs. Linux vs. Windows), Kernel structure and native level programming ,Runtime (Objective-C vs. Dalvik vs. Win RT), Security. Introduction to Android Operating System, Overview of android stack, Android features. Linux Kernel, Libraries, Android Runtime, Application Framework, Dalvik VM, Multithreading in Android-Async Task, Handler Post

UNIT II: Android Components: [15 HRS]
Activities, Services, Broadcast Receivers, Creating Broadcast receiver, Receiving System Broadcast, Understanding Broadcast action, category and data, Sending Broadcast. Content Providers, Views, layouts and Common UI components, Creating UI through code and XML, Activity life cycle, Intents-,Intent Filters, Intent-matching rules, Filters in your manifest. Communicating data among Activities. Selection components (Grid View, List View, Spinner), Adapters, Custom Adapters, Menus, Toast, Custom Toast, Dialogs, Status bar Notifications. Overview of services in Android, implementing a Service, Service lifecycle, Inter Process Communication (AIDL Services). Web Services and WebView - Consuming web services, Receiving HTTP Response (XML, JSON), Parsing JSON and XML, Using WebView.

UNIT II: Data Storage: [15 HRS]

Shared Preferences, Android File System, Internal storage, External storage. SQLite- Introducing SQLite, SQLite Open Helper and creating a database, Opening and closing a database, working with cursors, inserts, updates, and deletes.

Content Provider-Accessing built in content providers, Content provider MIME types, searching for content, Adding, changing, and removing content, creating content provider, Working with content files.

REFERENCES:

MANDATORY:

1. Lee, W. M. (2010). Beginning iPad application development. John Wiley & Sons.
2. Satya. K, Pro Android 4 ; Dave MacLean (Apress).
3. Burnette, E. (2009). Hello, Android introducing Google's mobile development platform 2nd.
4. Cinar, O. (2012). Android apps with Eclipse. Apress.
5. Dimarzio, J.F. Android- A Programmer's Guide (Tata McGraw Hill)

WEB BASED:

1. http://developer.android.com/index.html
2. http://www.appinventor.org/

Practicals: Mobile Application Development
Credit: 1
Marks: 25
Duration: 30 Hrs

1. Getting Started with Android – Installing the Development Environment, Configuring Android Stack.
2. Creating the First Android Application - Creating a Simple Android Project, Debugging Application through DDMS. Setting up environment. AVD Creation, Executing Project on Android Screen.
3. Android application development - Use of GUI components to implement a simple application such as a Calculator.
4. Review the earlier application making use of the advanced UI components.
5. Implementing Data storage application - an application to make Insert ,update, Delete and retrieve operation on the database.

Course Title: Server Side Programming
Course Code: COM-E7
Marks: 75
Credits: 3
Duration: 45 Hrs

Course Prerequisites: Nil

Course Objectives:
- Provide an in depth understanding of :
- Object oriented approaches to software development, in particular to the development of web applications.
- Server side scripts and their purpose

Course Outcomes:
Upon completion of the course students will be able to:

CO1: Design and implement server-side scripts.
CO2: Create and manipulate databases using SQL and server side technologies.
CO3: Create data documents using XML/JSON.
CO4: Enhancing web applications using AJAX and XML/JSON.
CO5: Develop dynamic web applications using the object oriented paradigm.

SYLLABUS:

UNIT I: Application of Object Oriented Approach to Software Development [10 HRS]
Review of Object Oriented Concepts - Class, Object, member variable, member function, Inheritance, Polymorphism, overloading, Data Abstraction, Encapsulation. Object oriented design and modelling.

UNIT II: Developing dynamic web applications with Server-side technologies [20HRS]
Static vs. Dynamic web pages ,Need for Server Side technologies, Multitier Web Architecture. Common Gateway Interface standard, server-side includes, server APIs, Server-side scripting – working principles, and implicit objects. Database and file access. Comparison of Web servers.

UNIT III: Enhancing Web Applications with Ajax and XML/JSON [15HRS]
AJAX – introduction, purpose, advantages and disadvantages. Key elements of AJAX –
introduction to XML. XML processing with server side script. XSL, transforms and templates. The XML Http Request object–methods and properties. Creating and using XML Http Request objects. Using XSLT with AJAX. JSON – Syntax, mixing literals, Array, object, encoding/decoding, JSON versus XML, server-side JSON tools.

Web Services - Introduction, its role. Basics – Creating, Publishing, Testing and Describing a Web services (WSDL)-Consuming a web service. SOAP – introduction, requests and responses. Role of UDDI – accessing registries. REST based webservices – building, deploying and consuming

REFERENCES:

MANDATORY:
1.Steelman, Murach'S Java Servlets & JSP, 2/E, PHI

SUPPLEMENTARY:
1. Chappell.D, Jewell.T, “Java Web Services : Using Java in Service Oriented Architectures”, O’Reilly
2. DT Editorial Services, (2018). Web Technologies, Black Book, Dream Tech

WEB BASED:
1. https://docs.oracle.com/cd/E14571_01/web.1111/e13712/basics.htm#WBAPP117

Practicals: Server Side programming
Credit: 1
Marks: 25
Duration: 30 Hrs

List of suggested PRACTICALS (the numbers in brackets indicate number of practicals)
1) Perform OOAD of a given system using the following diagrams: [3 P]
a) use case diagram b) class diagram
2) Using server side programming and following OOAD principles develop a dynamic web application. [6 P]
3) Add AJAX and Web service(s) to the application. [3 P]

Course Title: Human Computer Interface
Course Code: COM-E8
Marks: 75
Credits: 03
Duration: 45 Hours

Course Prerequisites: Nil

Course Objectives:
- To study the different aspects of human computer interaction.
- To study computer interface design concepts.

Course Outcomes:
Upon completion of the course student will be able to:

CO 1: To understand the intricacies of human interaction with a computer System.
CO 2: To understand the concept of a graphical user interface, and its design characteristics.
CO 3: To recognize the human element its strengths and weakness for computer interaction.
CO 4: To know the principles of good screen design and layouts.
CO 5: To know the different navigation schemes on windows-based interface; learn the different types of selection devices and components of a window-based interface.
CO 6: To know the different types of interaction devices and media.

SYLLABUS:

UNIT I: Introduction to Human Computer Interaction: [10 HRS]
Human-Computer Interaction, Evaluating Designs, The Birth of HCI. Importance of user Interface, Importance of good design, Benefits of good design, principles & heuristics of good design.
Importance of: Human characteristics, Human consideration, Human interaction speeds, Understanding business functions. User cantered design- Need-finding: Participant Observation, Interviewing, Additional Need finding, contextual inquiry & persona.

UNIT II: Rapid Prototyping and Graphical Interface Design: [20 HRS]
Rapid Prototyping: Story boarding. Paper Prototyping and Mock up, Video Prototyping, Creating and Comparing Alternatives.

Direct Manipulation. Mental Models. Heuristics (guidelines) for design.

Graphical Interface Design: Graphical user interface, standards such as Microsoft windows HCI guidelines, Windows: Navigation schemes selection of window; Selection of devices based and screen-based controls, Components, Text and messages, Icons, Multimedia, Colours., controls, help & error messages design.

UNIT III: Heuristic Evaluation and Visualization [15 HRS]
Web user interface design – Jessy James Garette five layers of user experience. Heuristic Evaluation: Heuristic Evaluation — Why and How?
Visualization, Amount of information, Focus and emphasis, Presentation information simply and meaningfully, Information retrieval on web, Statistical graphics.

REFERENCES:

MANDATORY:

1. Cooper, A., Reimann, R., & Dubberly, H. (2003). About face 2.0: The essentials of interaction design. John Wiley & Sons, Inc.
2. Alan.D, Janet.F, Gregory D. and Russell,B. (2012) Human-Computer Interaction, Prentice Hall.

SUPPLEMENTARY:

1. Shneiderman, B., & Plaisant, C. (2010). Designing the user interface: strategies for effective human-computer interaction. Pearson Education India.
2. Donald .A.N. (2010) The Design of Everyday Things Basic Books.

WEB BASED:

1. http://hcibib.org/
2. https://www.tutorialspoint.com/human_computer_interface/index.htm.
3.https://www.academia.edu/4955516/Wiley_The_Essential_Guide_to_User_Interface_Design_3rd_Edition_Apr_2007?auto=download.
4. https://www.slideshare.net/busaco/hci-2015-110-humancomputer-interaction-overview?qid=1c116f30-ec87-4eb4-a375-49b2bbe65d75&v=&b=&from_search=2

Practicals: Human Computer Interface
Credit : 1
Marks : 25
Duration: 30 Hrs

Suggested list of practical (Numbers in brackets indicate number of practicals)

1. Paper Prototyping using templates	(1)
2. Conducting survey interview and summarizing the result	(1)
3. Persona- conducting contextual interview and developing persona	(1)
4. GUI design- form design, menu design, help, error messages	(2)
5. Web UI design- pages, navigation, controls, Page submission – Asynchronous	(2)
6. Report designs	(2)
7. Visualization and info graphics	(1)
8. Heuristic evaluation	(2)
9. Story boarding	(1)

SEMESTER V

Course Title: Operating Systems
Course Code: COM-V.C-7
Marks: 75
Credits: 3
Duration:45 Hrs

Course Prerequisites: Nil

Course Objectives:

- To understand different functions of an operating system.
- To study various aspects of operating system like Process Management, Memory Management, Storage management etc.
- To understand different algorithms used for CPU scheduling, Memory allocation.

Course Outcomes:
Upon completion of the course students will be able to:

CO1: Identify the functions of operating system.
CO2: Describe process management and process coordination.
CO3: Implement various CPU scheduling algorithms and compare their performance.
CO4: Explain deadlock handling and recovery from deadlock.
CO5: Implement various Memory Allocation algorithms.
CO6: Describe implementation of Virtual memory.
CO7: Describe storage management and storage structure.

SYLLABUS:

UNIT I : Introduction to Operating System and Process Management [15HRS]
Introduction to Operating System: Basic elements of a computer system: Processor, Main Memory, I/O Modules, System Bus, Instruction Execution; Operating Systems: Definition, Operating system Structure, operating system operations, Relationship between Kernel, OS, and Hardware, Operating system services, System calls, Types of system calls, System programs.
Process Management**:** Process Definition, Process Control Block, Process States, Operations on Process; Inter process communication, Threads and Microkernels: Definition, Multi-threading Model Process Scheduling, Scheduling Criteria, Scheduling Algorithms, Multi-Processor Scheduling.

UNIT II : **Process Coordination [15HRS]**
Process Coordination

Process Synchronization, Principles, Mutual Exclusion, The Critical-Section Problem, Petersons Solution, Semaphores, Monitors, Readers/Writers Problem; Classic Problems of Synchronization, Dining Philosopher's problem.

Deadlocks- system models, Deadlock characterization, Deadlock Handling Methods, Prevention, Avoidance, Detection, Recovery From Deadlock.

UNIT III: Memory Management and Storage Management [15HRS]

Memory Management: Introduction, Swapping, Contiguous Memory Allocation, Paging, Page Table, Segmentation, Virtual Memory: Introduction, Demand Paging, Page Replacement, Allocation of Frames, Thrashing
Storage Management : File System, Concepts, File Organization and Access Methods, Directory and Disk Structure.
Secondary Storage Structure : Overview, disk structure, Disk attachment, Disk scheduling
Reading.

MANDATORY:

1. Silberchatz, A., Galvin, , & Gagne. (2008). Operating System Concepts (8th ed.). Wiley publication.

SUPPLEMENTARY:

1. Stallings, W. (2001). Operating systems: Internals and design principles.(6th ed.). Upper Saddle River, N.J: Prentice Hall.

WEB BASED:

1. https://www.geeksforgeeks.org/operating-systems/
2. Video Links and Animations
3. https://www.youtube.com/watch?v=WP3uDglbPiI
4.http://williamstallings.com/OS-Animation/Animations.html
5.Linux Tutorials for Practical
6.https://www.tutorialspoint.com/unix/index.htm

Practical : Operating Systems
Credit : 1
Marks : 25
Duration:30 Hrs

Any 8 from the following can be done.

1. Demo/Review of Installing Linux / Windows Operating System, Partitioning and formatting disk, Installing applications device drivers, working with files, mounting file systems, checking system space, creating, modifying and deleting user accounts.
2. Study of Basic commands of Linux.
3. Shell Programming in Unix/Linux, arithmetic operations, loops.
4. Shell Programming – advanced.
5. Menu Driven Shell scripting.
6. Filters and Pipes in LINUX.
7. Implementation of Inbuilt Linux/UNIX commands like cp, rename etc.
8. Implementation of CPU scheduling policies.
9. Implementation of Memory allocation techniques.
10. Implementation of Banker's algorithm. (Resource Allocation Graph).

Course Title: Embedded Systems
Course Code: COM-V.E-9
Marks: 75
Credits: 3
Duration: 45Hours

Course Prerequisites:

- Knowledge of Programming

Course Objectives:

- To have a thorough understanding of major components of Embedded Systems.
- Implement small modules of programs to solve well defined problems in an Embedded Systems.
- To familiarize with the tools to develop applications in an Embedded Environment.

Course Outcomes:
Upon completion of the course students will be able to:

CO1: Describe Embedded Systems and its characteristics.
CO2: Classify the Embedded processors and their design metrics.
CO3: Summarize the performance of ARM processors and various components of Embedded Systems.
CO4: Classify Sensors and Actuators and identify their functions and applications.
CO5: Categorize I/O devices, I/O Interfacing and Communication protocols along with their functions.
CO6: Generalize the functionality of IoT and RTOS.
CO7: Design and develop Embedded / IoT Applications using Arduino/Raspberry-Pi boards.

SYLLABUS:

UNIT I: Introduction to Embedded Systems and System Hardware [15 HRS]
Introduction to Embedded Systems, Microprocessors and Micro-controllers. Components of Embedded System & its Classification, Characteristic of embedded system. Introduction to embedded processor, Digital Signal Processor, Application Specific System Processor, Design Process in Embedded System, Design metrics, Steps in design process. Challenges in Embedded System design, Design Examples. Advances in Embedded Systems.

System hardware, Interrupt structure and Applications, ARM Processor - Architecture, Programmer's model, Modes of operation, Interrupt**,** Handling Interrupts, Comparison of ARM7 & ARM9.
Components of Embedded Systems-Management of Power Supply, Clocking Unit, Real Time Clock and Timers, Reset Circuitry and Watchdog Timer. Processor and Memory Selection, Memory Map of Embedded System, Interfacing Processors, Memories and I/O – Analog vs Digital. Overview of Arduino, Intel Edison and Raspberry Pi boards.

UNIT II: Input/output Devices and Internet of Things [15 HRS]
I/O interfacing and Communication Buses, Serial vs Parallel Communication, Serial Data Communication RS-232/UART.
I/O devices, ADC / DAC, Optical Devices such as LED / LCD Display devices, Opto-Isolator, Relays, DC motor, Stepper motor, Timers/Counters, Parallel ports - Device interfacing. Serial communication

Protocols - UART Protocols, I2C, CAN, USB & ZigBee – Protocol Architecture, Topology, Packets, Communication Cycle, Arbitration, Applications and comparison.
Introduction to IoT, M2M to IoT-The Vision-Introduction, M2M towards IoT- the global context, IoT Architectural Overview, Potential and Challenges.

UNIT III: Sensors, Actuators and RTOS [15 HRS]

Sensors / Transducers: Principles – Classification – Parameters – Characteristics –Environmental Parameters (EP) – Characterization.

Thermal Sensors: Introduction – Gas thermometric Sensors – Thermal Expansion Type Thermometric Sensors.

Magnetic sensors: Introduction – Sensors and the Principles Behind – Magneto-resistive Sensors, Semiconductor Magneto resistors.

Smart Sensors: Introduction – Primary Sensors – Excitation – Amplification – Filters – Converters – Compensation – Information Coding/Processing - Data Communication – Standards for Smart Sensor Interface – The Automation.

Sensors Applications: Introduction – On-board Automobile Sensors (Automotive Sensors) – Home Appliance Sensors – Aerospace Sensors –– Sensors for Manufacturing – Sensors for environmental Monitoring.

Actuators: Overview of Pneumatic and Hydraulic Actuation Systems, Mechanical Actuation Systems, Electrical Actuation Systems.

RTOS: Introduction to RTOS, architecture of kernel, task and task scheduler, interrupt service routines in RTOS Environment.

REFERENCES:

MANDATORY:

1.Rajkamal, (2010), Embedded Systems – Architecture, Programming and Design, (2nd ed.), Tata McGraw Hill.
2.D. Patranabis, (2013), Sensors and Actuators, (2nd ed), PHI.

SUPPLEMENTARY:

1. David E Simon, (2004),An Embedded Software Primer, (1st ed.), Pearson India.
2. Tammy Noergaard, (2012), Embedded Systems Architecture, (2nd ed.), Elsevier
3. D. Patranabis, (2010),Sensors and Transducer, (2nd ed.), PHI Learning Private Limited.
4. Jan Holler, VlasiosTsiatsis, Catherine Mulligan, Stefan Avesand, Stamatis Karnouskos, David Boyle, (2014), From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence, (1st ed), Academic Press,.

WEB BASED:

1.https://www.tutorialspoint.com/internet_of_things/index.htm
2.https://www.arduino.cc/
3.http://www.zdnet.com/article/m2m-and-the-internet-of-things-a-guide
4.https://www.raspberrypi.org/
5.www.thingspeak.com
6.https://blynk.io/

Practicals: Embedded Systems
Credit: 1
Marks: 25
Duration: 30 Hrs

Programs to be executed on some of the embedded boards like Arduino, Intel Edison, Raspberry Pi, Bolt, etc that covers the following tasks (Any 10 experiments):

1) Interfacing sensors (3P)
2) Interfacing output devices (1P)
3) Interfacing input devices (1P)
4) Interfacing actuators (1P)
5) Building obstacle avoiding Robot (1P)
6) Line Following Robot (1P)
7) Programming with Raspberry Pi (2P)
8) Monitoring Data over Cloud (1P)
9) Building Web app to control devices (1P)
10) Mini Project.

Course Title: Data Base Management System - II
Course Code: COM-V.E-17
Marks: 75
Credits: 3
Duration:45 Hrs

Course Prerequisites:

- Knowledge of Data Base Management Systems.

Course Objectives:

 To provide advance database solutions.

Course outcome:

Upon completion of the course students will be able to:

CO1: Formulate complex queries for database updation.
CO2: Implement stored procedures and Functions.
CO3: Understand concurrent transactions and Recovery mechanisms.
CO4: Develop a full database application.
CO5: Understand and appreciate the NoSQL databases.

SYLLABUS:

1. Advanced SQL [15HRS]

SQL data types and schemas, Integrity constraints, Authorization, Embedded SQL, Dynamic SQL, Triggers, Stored Procedures, views

2. Indexing and Hashing [7HRS]

Basic concepts, Ordered Indices, Dense and Sparse Indices. B and B+ trees Hashing – Static hashing, Dynamic Hashing, Extendable hashing, Comparison of Ordered Indexing and Hashing.

3. Transaction, Concurrency Control, Recovery System. [10HRS]

Transaction: Transaction concept, Transaction state, Implementation of Atomicity and Durability, concurrency. Serializability, conflict serializability. Concurrency Control : Lock-Based Protocol Recovery System: Failure Classification, Storage structure, Stable storage implementation, Recovery and Atomicity: Log-Based Recovery.

4. Introduction to Big data and NoSQL

[13HRS]Introduction to the Big Data problem. Current challenges, trends, and applications Comparison between SQL and NOSQL Databases Types and examples of NoSQL databases- Column, Document, Key-value, Graph, Multi-model. Introduction to Document type NoSQL database such as Mongo Db. - Introduce concepts of collection and documents, Advantages, Data types, Projections, indexing, Sharding .

MANDATORY:

1.A Silberschatz, H F Korth, S Sudarshan, Database system concepts, McGraw-Hill ,sixth Edition

SUPPLEMENTARY:

Ramakrishan, J Gehrke, ―Database management systems", McGraw-Hill , 3rd edition R Elmasri, S B Navathe, ―Fundamentals of database Systems", Pearson Education , 5th Edition Kristina Chodorow MongoDB : The Definitive Guide (English) O'Reilly 2nd Edition.

Practicals: Database Management Systems II
Credits: 1
Marks: 25

1. SQL Revision

2. Advance SQL- Dynamic SQL, Triggers Advance SQL- Stored Procedures

3. Using ODBC API for insertion of record into database. Using ODBC API for deletion of record. Using ODBC API for modification of data. Using ODBC API for data retrieval.

4. Installing and Creating a document using MongoDb concept Performing Indexing using MongoDb.

5. Performing aggregation functions on MongoDb Implementation of Master-Slave approach.

6. Connection of Mongo Db using Java Insertion, modification, deletion using Mongo Db Data retrieval using Mongo Db Shrading using Java and Mongo Db.

7. Mini Project

Course Title: Web Development Framework
Course Code: COM-VI. E-18
Marks: 75
Credits: 03
Duration: 45 HRS

Course Prerequisites:

- Knowledge of Web Design, Object Oriented Paradigm and Database Management System.

Course Objective:-

- Use Web Frameworks and Libraries to develop interactive web applications.

Course Outcomes:
On completion of the course student will be able to:

CO1: Use ReactJS to build rich and interactive front end applications.
CO2: Use NodeJS to develop back end application to accept POST, GET, PUT, DELETE requests.
CO3: Develop REST API's using NodeJS.
CO4: Write non-blocking and blocking JavaScript code.
CO5: Explain Framework and Libraries with respect Web Development.

UNIT I **[15HRS]**

ReactJS

History of front end libraries, Motivation for using React, Thinking in React, One way binding, JSX + CSS modules, Virtual DOM, ES6

ReactJS components
Component life cycle, Component API, Render functions, State, Props, Mix ins.

UNIT II **[15HRS]**

ReactJS : Interaction between components

Passing data from parent to child, Passing data from child to parent, Passing data between 2 components at the same level, Forms, Refs, React-Router, API integration.

NodeJS: Introduction. Brief overview on the benefits of using Node.js and how Node.js is used in modern web development, Node and NPM, Introduction to setting up a Node.js project, Importing modules using npm, Using core modules to make HTTP requests and manipulate the file system.

UNIT III **[15HRS]**

NodeJS: Express framework

Set up a web server, Implementing API routing, Implementing middle-ware, Implementing URL parameters.

NodeJS: MySQL module

Setting up a database and connecting it to a Nodel server ,Storing and retrieving data from the database.

REFERENCES

MANDATORY:

1.Brett McLaughlin (2011). What Is Node?(1st ed) O'Reilly Media
2.Alex Banks (2017).Learning React.(1sted) Shroff /O'Reilly.

REFERENCES:

1. Mario Casciaro (2016). Node.js Design Patterns (2nd ed) Packt Publishing Limited

WEB BASED:

1. https://www.tutorialspoint.com/nodejs/index.htm
2. https://reactjs.org/docs/getting-started.html
3. https://www.youtube.com/watch?v=Ke90Tje7VS0
4. https://ict.iitk.ac.in/node-js-books/
5. https://freefrontend.com/nodejs-books/

Practical: Web Development Framework
Marks: 25
Duration: 30 hrs
Credits: 01

List of Practicals

1. Creating a simple web server. (1P)
2. Connect to MySQL database. (1P)
3. CRUD using MySQL database API's. (4P)
4. Fetchdatafromaform,validateandinsertinthedatabase.Deletedatainthedatabase.
5. Updated in the database.
6. Display data from the database.
7. Uploading files. (1P)
8. Login functionality using sessions. (1P)
9. Using cookies to store website data. (1P)
10. Completion of Mini project.

Course Title: Software Testing
Course Code: COM-V.E-12
Marks: 75
Credits: 3
Duration:45 Hrs

Course Prerequisites:

- Knowledge of Software Engineering

Course Objectives:

- To understand the fundamental concepts in software testing
- To study the various software testing strategies
- To learn how to design test cases and execute them.

Course Outcomes:

Upon completion of the course students will be able to:

CO1: Understand the different software testing strategies.
CO2: Apply testing strategies to live projects.
CO3: Design test cases.
CO4: Execute test cases using software testing tools.

SYLLABUS:

UNIT I: **[15HRS]**

Software testing principles - Software Testing- Need for testing, Psychology of testing ,Testing economics, SDLC and Testing, Verification & Validation. Quality Assurance, Quality Control.

Testing strategies and types - White box testing techniques - Statement coverage, Branch Coverage , Condition coverage, Decision/Condition coverage , Multiple condition coverage ,Dataflow coverage, Automated code coverage analysis, Inspections, Walkthroughs Code Review Black box testing techniques - Boundary value analysis, Robustness testing ,Equivalence partitioning, Syntax testing, Finite state testing, Levels of testing, Unit, Integration and System Testing, Compatibility Testing, Domain Testing, Adhoc Testing ,Use of Requirement, Traceability Matrix.

UNIT II: **[20HRS]**

Integration Testing Waterfall - Top-down ,Bottom up ,Big bang, Sandwich System and Performance Testing - Types of system testing ,Functional and non-functional testing Acceptance Testing ,Setting entry and exit criteria for phases and typical product release scenarios ,Basic factors governing performance testing, Methodology for performance testing, Tools for performance testing.

Regression Testing - Purpose ,Timing, Choice of tests ,Smoke tests ,Best practices Internationalization and Localization testing - Preliminary concepts, Adhoc testing, Pair testing, Extreme testing, Agile testing, Exploratory testing, Defect seeding.

Usability Testing - Factors in usability testing, Aesthetics testing, Accessibility testing, Tools for usability testing.

Testing object oriented software - Definitions and Challenge differences from testing non-OO Software, Class testing strategies Class Modality, State-based Testing, Message Sequence Specification.

UNIT III: [10HRS]

People and organizational issues in testing - Common people issues and myths in testing, Providing career paths in testing, Organizational structures for testing teams, Geographically distributed testing teams and success factors.

Test Management and Automation- Test
Planning, Test Management, Test Process, Test Reporting, Test Automation, Factors to consider in automation, Challenges in test automation, Test Metrics, Product Metrics, Process Metrics, Progress Metrics. Use of metrics in ascertaining product release.

REFERENCES:

MANDATORY:

1. Srinivasan D.,Gopalaswamy R.;(2009) Software Testing- Principles and Practices, 4th Edition: Pearson Publication.

SUPPLEMENTARY:

1. Jalote P., (2010)An Integrated Approach to Software Engineering, 3rd Edition :Narosa Publishing House
2. Pressman R., (2017) ,Software Engineering: A Practitioners Approach, 6th Edition :McGraw Hill Publication.

WEB BASED:

1.https://www.guru99.com

Practicals: Software Testing
Credit: 01
Marks: 25
Duration: 30 Hrs

1. Planning Test Cases
2. Generating Test Cases/Test Suite
3. Enhancing Tests
4. Debugging Tests
5. Running Tests
6. Analysing Results
7. Reporting Defects

SEMESTER VI

Course Title: Computer Networks
Course Code: COM-VI. C-8
Marks: 75
Credits: 3
Duration: 45 Hrs

Course Prerequisites: Nil

Course Objectives:

- To understand the basic concepts of Computer Networking.
- To understand the layered architecture of computer networks.
- To understand various transmission media used for networking.
- To understand working of various protocols in different layers.

Course outcomes:
Upon completion of the course students will be able to:

CO1: Understand the need for Network and various layers of OSI and TCP/IP reference model.
CO2: Explain various Data Communications media.
CO3: Identify the different types of network topologies and Switching methods.
CO4: Describe various Data link Layer Protocols.
CO5: Identify the different types of network devices and their functions within a network.
CO6: Differentiate between various Classless and Class full IP addresses with Sub neting concept.
CO7: Analyze and interpret various Network and Transport Layer protocols.
CO8: Explain different application layer protocols.

SYLLABUS:

UNIT I **[20HRS]**

Introduction
Basics of Computer Networks, Classification: transmission technology, scale; Applications; Data Communications: data, signal, bandwidth, bit interval and bit rate, Modes of Communication. Layered network architecture, Networks models: OSI model, TCP / IP protocol suite; Guided and Unguided Transmission media, Multiplexing: FDM, TDM. Switching: Circuit switching, message switching, Packet Switching.

Data link layer
Data link control: Framing: Character Count, Character Stuffing, Bit Stuffing; , Error Detection and correction, Flow and error control, HDLC; Multiple access: Random access – Controlled access, ALOHA, CSMA, CSMA/CD and CSMA/CA; Ethernet : IEEE standards, standard Ethernet, Fast Ethernet, Gigabit Ethernet.

UNIT II **[15 HRS]**

Network layer

Connecting devices: repeater/hub, bridge, router and gateway, Backbone networks - Virtual LANS. Functions of Network layer; Network Service types: Virtual Circuits, Datagrams; Logical addressing: IPv4, private and public IP addressing, special IP addresses, subnetting, IPV6 addressing Internet Protocol: Internetworking:IPv4, Fragmentation and reassembly , Address mapping : ARP, RARP, BOOTP, DHCP, ICMP . Routing: classification of routing, Shortest path routing, Distance Vector routing, Link State routing;

UNIT III: Transport layer, Application layer and Wireless network [10HRS]

Transport layer**:** Process-to-Process delivery: User Datagram Protocol (UDP), Transmission Control Protocol (TCP), Quality of services (QoS)

Application Layer: Domain Name System (DNS), E-mail, FTP, HTTP.

Basics of Wireless Networking

REFERENCES:

MANDATORY:

1. Tanenbaum A., W.(2010) Computer Networks (5th ed.) . Pearson Education.

SUPPLEMENTARY:

1. Behrouz A. Forouzan B.(2017) Data communication and Networking(5th ed.). McGraw Hill Education
2. Kurose J, K.(2017) Computer Networking – A Top-Down Approach (6th ed.) Pearson Education,
3. Mir N. (2006) Computer and Communication Networks, Prentice Hall Publishers.

WEB BASED:

1.https://www.youtube.com/watch?v=tj7f244tubM
2.https://www.youtube.com/watch?v=vFypCugyFoM
3.https://www.geeksforgeeks.org/computer-network-tutorials/
4. Slides of the Book . Andrew S. Tanenbaum, David J. Wetherall“Computer Networks”, Prentice-Hall, 5th Edition.:https://www.pearson.com/us/higher-education/product/Tanenbaum-Power-Point-Lecture-Slides-for-Computer-Networks-5th-Edition/9780132127066.html?tab=downloadable-resources
5.https://www.tutorialspoint.com/data_communication_computer_network/index.htm
6.https://www.cisco.com › training-events › netacad › course _catalog › docs
Slides for the book Behrouz A. Forouzan, “Data communication and Networking”, Tata McGraw – Hill, 2011, 4thEdition: 8.http://www.mhhe.com/engcs/compsci/forouzan/dcn/index.mhtml

Practicals: Computer Networks
Credits: 1
Marks: 25
Duration: 30 Hrs

Practical (Any 6 practical):

1. Setting up of LAN Network (2P)
2. IP address manipulation -Extract network id and Host id given netmask (2P)/Mini Project
3. Configuring routing tables
4. TCP Socket programming (2P)
5. UDP Socket programming (2P)
6. Mini Project / Simulation of IP fragmentation
7. Mini Project/Configuring E-Mail/DNS
8. Installing virtual machines, Ethernet cabling

Course Title: Network Security
Course Code: COM-VI. E-13
Marks: 75
Credits: 3
Duration: 45 Hrs

Course Prerequisites:

- Knowledge of Programming

Course Objectives:

- To understand the theory and concepts of Network Security.
- To be able to secure a message over insecure channel by various means.
- To learn about how to maintain the Confidentiality, Integrity and Availability of a data.

Course Outcomes:

Upon completion of the course students will be able to:

CO1: classify the symmetric encryption techniques
CO2: Illustrate various Public key cryptographic techniques
CO3: Evaluate the authentication and hash algorithms.
CO4: Discuss authentication applications.
CO5: Summarize the intrusion detection and its solutions to overcome the attacks.
CO6: Basic concepts of system level security

SYLLABUS

UNIT I: **[15HRS]**

Concepts of Security & Classical Encryption Techniques:
Introduction, The need for security, Security Approaches, Security Attacks, Security Services, Security Mechanisms, A Model for Network Security, Classical Encryption Techniques :Substitution techniques, Transposition techniques, Steganography.

Design Principle of Block Cipher:
Block Cipher Operation: Electronic Code Book, Cipher Block Chaining, Cipher Feedback, Output Feedback, Counter, Feistel Cipher, The Data Encryption Standard.

Cryptography:
Mathematical Tools
Introduction to Number Theory, Modular Arithmetic, Prime Numbers, Euler‘s Totient Function.

UNIT II: **[15HRS]**

Public Key Cryptography:
Principles of Public Key Cryptosystems, The RSA Algorithm, Other Public key cryptosystems, Diffie Hellman Key Exchange.

Cryptographic Hash Functions:
Applications of Cryptographic Hash Functions, Two Simple Hash Functions, Hash Functions Based on Cipher Block Chaining, MD5 Message Digest Algorithm, Secure Hash Algorithm SHA 512.
Message Authentication Codes and Digital Signatures:
Message Authentication Requirements – Message Authentication Functions –Requirements for Security of MACs ,MACs Based on Hash Functions, HMAC, MACs Based on Block Ciphers, Data Authentication Algorithm. Digital Signatures, Digital Signature Standard.

UNIT III: [15 HRS]
Key Management & Distribution And User Authentication:
Introduction, Digital Certificate, Private key Management, The PKIX Model, Public key cryptographic standards ,XML, PKI and security.

Program Security:
Flaws, Malicious code: viruses, Trojan horses, worms, Program flaws: buffer overflows, time-of check to time-of-use flaws, incomplete mediation.

Firewall and Virtual Private Network:
Introduction to network security techniques: IP Security, firewalls, virtual private networks.

REFERENCES:

MANDATORY:

1.William.S,(2017).Cryptography and Network Security – Principles and Practices(7th ed.).Prentice Hall of India.

SUPPLEMENTARY:

1.Charles P. Pfleeger and Shari L. Fleeger(2015).Security in Computing(7th ed.). Prentice-Hall.
2.Atul.K,(2007).Cryptography and Network Security(2nd ed.).Tata McGraw-Hill.
3.Menezes A. J., P.C. Van Oorschot and S.A. Vanstone(1997) ―Handbook of Applied Cryptrography(2nd ed.).Jaypee medical.

WEB BASED:
1. Cryptography and Network Security - Nptel:
https://nptel.ac.in/courses/106/105/106105031/
2. William Stallings, ―Cryptography and Network Security – Principles and Practices‖, Prentice Hall of India:
http://www.amorena.com.ar/PAG%20DE%20MATERIAS%20Y%20LIBROS/LIBROS%20TODOS/CRYPTOGRAPHY%20AND%20NETWORK%20SECURITY,%20PRINCIPLES%20AND%20PRACTICE.pdf
3. Cryptography Tutorial:
https://www.tutorialspoint.com/cryptography/index.htm
4. IPSec, VPN, and Firewall Concepts:

http://www.cs.unh.edu/~it666/reading_list/Networking/firewall_concept_terms.pdf
5. Implementation of Hill Cipher:
https://www.tutorialspoint.com/cplusplus-program-to-implement-the-hill-cypher

Practicals: Network Security
Credits: 1
Marks: 25
Duration: 30 Hrs

1. Implementation of Caesar Cipher
2. Implementation of One-Time Pad
3. Implementation of Playfair Cipher
4. Implementation of Hill Cipher
5. Implementation of Data Encryption Standard Algorithm
6. Implementation of Image Steganography
7. Implementation of RSA Algorithm
8. Implementation of Digital Signatures using RSA Algorithm
9. Design Network protocol analyser tool to analyse network traffic.
10. Mini Project/ Case Study

Course Title: Cloud Computing

Course Code: COM-VI. E-14

Marks: 75

Credits: 3

Duration:45 Hrs

Prerequisite Courses:

- Operating Systems(COM-V.C-7)

Course Objectives:

- To make students understand the key elements of cloud computing.
- To understand the difference between deploying applications on the cloud and the local infrastructure.
- To understand various cloud service models.

Course Outcomes:

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

CO1: Explain the core concepts of the cloud computing paradigm.
CO2: Characterize the different cloud services ie. Infrastructure, Platform and Software as a Service (IaaS, PaaS, SaaS).
CO3: Deploy application in a production environment.
CO4: Host a cloud platform like Apache OwnStack and Owncloud.

SYLLABUS:

UNIT I: Introduction to Cloud Computing
[**15HRS**]

Recent trends in Computing- Grid Computing, Cluster Computing, Distributed Computing, Utility Computing, Cloud Computing. Cloud Computing definition, History of Cloud Computing, How Cloud Computing Works, 5-4-3 Principles of Cloud Computing, Cloud Ecosystem, Benefits and challenges of cloud computing, Issues for Cloud Computing.

UNIT II: Cloud Computing Architecture
[15HRS]

Comparison with traditional computing architecture (client/server), Cloud Architecture - Benefits of Cloud Architecture, Cloud Computing Architectural components. Cloud Computing Service Models- IaaS, PaaS and SaaS. Deployment Models- Public cloud, Private cloud, Hybrid cloud and Community cloud, Key drivers to adopting cloud, Impact of cloud on users, Governance in the cloud. Introduction to Virtualization, Different approaches to Virtualization, Hypervisors, Types of Virtualization, Machine Image, Virtual Machine(VM).

UNIT III: Cloud Service Providers
[15HRS]

Amazon Web Services (AWS)- Business and Technical benefits of Cloud Services, Elasticity, IAM, Security Groups, VPC, EC2, Traditional IP v/s Elastic IP, Amazon Machine Image, Elastic Load Balancing, CloudWatch, Elastic Block Storage, S3, SQS, SNS, RDS, Dynamo DB, Ops Works, Server less Architecture- API Gateways, Lambda. Hadoop- MapReduce, HDFS, Hadoop on AWS.
Microsoft Windows Azure: Azure Virtual Machines, Cloud Services, Application scenarios.

REFERENCES:

MANDATORY:

1. Chandrasekaran, K.. Essentials of Cloud Computing. United Kingdom, CRC Press, 2014.
2. Tim mather, subra kumarswamry and sharhed Latif, “Cloud Computing Security and Privacy”, O’Reilly publication.
3. Richard Hill, Laurie Hirsch, Peter Lake, Siavash Moshiri, “Guide to Cloud Computing Principles and Practices”, Springer.
4. Buyya, R., Broberg, J., & Goscinski, A. M. (Eds.). (2010). Cloud computing: Principles and paradigms (Vol. 87). John Wiley & Sons.

SUPPLEMENTARY:

1. 1. Nikos Antonopoulos, Lee Gillam“Cloud Computing: Principles, Systems and Applications”, Springer.
2. Vines, R. L. K. R. D., & Krutz, R. L. (2010). Cloud security: A comprehensive guide to secure cloud computing (pp. 35-41). Wiley Publishing, Inc.

WEB BASED:

1. https://azure.microsoft.com/
2. https://aws.amazon.com/what-is-cloud-computing/
3. https://cloud.google.com/appengine
4. https://www.ibm.com/cloud/learn/cloud-computing
5. https://www.salesforce.com/in/learning-centre/tech/cloudcomputing/

Practicals: Cloud Computing
Credit: 1
Marks: 25
Duration: 30 Hr

1. Virtualization
   [3P]
   a. Install Oracle VirtualBox/VMware and create two VMs on your laptop.
   b. Install VirtualBox/VMware Workstation with Linux and windows OS on top of windows.
   c. Install a C compiler in the virtual machine created using virtual box and execute Simple Programs
   d. Find a procedure to transfer the files from one virtual machine to another virtual machine.
   e. Test ping command to test the communication between the guest OS and Host OS
2. Docker container
   [3P]
   a. Installation and Setup of Docker container Linux/Windows
   b. Containerizing the Python/Java/Node.js application
   c. Creating a persistent Data Base.
   d. Logging into Docker Hub
   e. Pushing the container to the Docker hub.
   f. Running the application from the container in pushed Docker hub
3. Own Cloud(Iaas)
   [2P]
   a. Installation and setup of Own Cloud

b. Explore the features of Own Cloud
c. Create, Manage and groups User accounts in own Cloud by Installing Administrative Features.
d. Federated sharing

4. Google App Engine.
   [2P]
   a. Install and configure Google app engine
   b. Create hello world app and other simple web applications using python/java.
   c. Use GAE launcher to launch the web applications.
5. Heroku(PaaS)
   [2P]
   a. Deploy a website on Heroku
   b. Integrate Postgresql and deploy website on heroku
6. Hadoop
   [3P]
   a. Install Hadoop single node setup.
   b. Develop hadoop application to count no of characters, no of words and each character frequency.
   c. Develop hadoop application to process given data and produce results such as finding the year of maximum usage, year of minimum usage
   d. Install Hadoop single node cluster and run simple applications like wordcount.

Course Title: Introduction to Data Science
Course Code: COM-V.E-11
Marks: 75
Credits: 3
Duration:45 Hrs

Course Prerequisites:

- Statistical Methods
- Basic probability and statistics.

Course Objectives:

- Become familiar with methods of data science and their practical usefulness.
- To learn, understand, and practice machine learning approaches.
- To analyse large and unstructured data with different tools.

Course outcomes:
Upon completion of the course students will be able to:

CO1: Describe what Data Science is and the skill sets needed to be a data scientist.
CO2: Explain in basic terms what Statistical Inference means. Identify probability distributions commonly used as foundations for statistical modelling. Fit a model to data.
CO3: Explain the significance of exploratory data analysis (EDA) in data science. Apply basic tools(plots, graphs, summary statistics) to carry out EDA.
CO4: Describe the Data Science Process and how its components interact.
CO5*:* Apply basic machine learning algorithms for predictive modelling.
CO6: Identify common approaches used for Feature Generation. Identify basic Feature Selection.
CO7: Reason around ethical and privacy issues in data science conduct and apply ethical practices.
CO8: Create effective visualization of given data (to communicate or persuade).
CO9: Use of Mining Social-Network Graphs in Data science out basic statistical modelling and analysis.

SYLLABUS:

UNIT I: **[10HRS]**
Introduction to Data Science
What is Data Science? Big Data and Data Science hype -and getting past the hype,
Why now? –Datafication, Current landscape of perspectives, Skill sets needed.

Statistical Inference:
Populations and samples, Statistical modelling, probability distributions, fitting a model,
Intro to R.

UNIT II: **[20HRS]**
Exploratory Data Analysis and the Data Science Process:

Basic tools (plots, graphs and summary statistics) of EDA, Philosophy of EDA, The Data Science Process, Case Study: Real Direct (online real estate firm).

Three Basic Machine Learning Algorithms:
Linear Regression, k-Nearest Neighbours (k-NN),k-means.

Feature Generation and Feature Selection (Extracting Meaning From Data):
Motivating application: user (customer) retention, Feature Generation (brainstorming, role of domain expertise, and place for imagination), Feature Selection algorithms, Filters; Wrappers; Decision Trees; Random Forests.

UNIT III: [15HRS]

Mining Social-Network Graphs:
Social networks as graphs, Clustering of graphs, Direct discovery of communities in graphs, Partitioning of graphs, Neighbourhood properties in graphs

Data Visualization:
Basic principles, ideas and tools for data visualization, Examples of inspiring (industry) projects, Exercise: create your own visualization of a complex data set.

Data Science and Ethical Issues:
Discussions on privacy, security, ethics, A look back at Data Science, Next-generation data scientists

REFERENCES:

MANDATORY:
1.O'Neil, C., & Schutt, R. (2013). Doing data science: Straight talk from the frontline. " O'Reilly Media, Inc."

SUPPLEMENTARY:
1. Jure.L,, Anand. R,Jeffrey.U(2014). Mining of Massive Datasets v2.1(2nd ed.).Cambridge University Press.
2. Kevin P. Murphy,(2012).Machine Learning: A Probabilistic Perspective .MIT Press.
3.Provost, F., & Fawcett, T. (2013). Data Science for Business: What you need to know about data mining and data-analytic thinking. " O'Reilly Media, Inc.".
4.Hastie, T., Tibshirani, R., & Friedman, J. (2009). The elements of statistical learning: data mining, inference, and prediction. Springer Science & Business Media..
5.Blum, A., Hopcroft, J., & Kannan, R. (2020). Foundations of data science. Cambridge University Press.
6. Zaki, M. J., Meira Jr, W., & Meira, W. (2014). Data mining and analysis: fundamental concepts and algorithms. Cambridge University Press.
7. Han, J., Pei, J., &Kamber, M. (2011). Data mining: concepts and techniques. Elsevier.

WEB BASED:

1. Cathy O’Neil and Rachel Schutt. Doing Data Science, Straight Talk From The Frontline, O’Reilly https://classroom.google.com/u/0/w/NDA4ODg4MTQ3MjZa/t/all
2. Data Science E-Learning Course:

https://onlinecourses.nptel.ac.in/noc19_cs60/unit?unit=5&lesson=6
3. Simple Linear Regression Example:
https://www.spss-tutorials.com/simple-linear-regression/
4**.** https://www.kaggle.com/pavansanagapati/a-simple-tutorial-on-exploratory-data-analysis
5. Data visualization
6. https://paldhous.github.io/ucb/2016/dataviz/week2.html
7. https://www.targetprocess.com/articles/visual-encoding/

Practicals : Introduction to Data Science
Credit: 1
Marks: 25
Duration: 30 Hrs

1. Implementation of probability distribution
2. Sampling and re-sampling.
3. Linear Models
4. K-Nearest neighbour
5. K-Means
6. Feature Selection Algorithm
7. Filters and Wrappers
8. Decision Trees

All the experiments will be implemented using Excel /R-Tool/ or equivalent.

Course Title: Web Development with Laravel
Course Code: COM-E-19
Marks: 75
Credits: 03
Duration: 45hrs

Course Prerequisite: Object Oriented Programming

Course Objectives:

1. Provide an in depth understanding of a server side language, and use it

to develop applications

2. Design and implement basic server-side scripts.

3. Create responsive and interactive web applications using frameworks

Course Outcomes:
On successful completion of this course students will be able to:

CO1: Explain the core features and functionalities of PHP

CO2: Design interactive web application using core PHP

CO3: Develop basic server side script to interact with

users and the database

CO4: Build a web application using laravel framework

CO5: Utilize MVC model

UNIT 1: [15 Hrs]

Static vs. Dynamic web pages, Need for Server Side technologies, Multi Tier Web Architecture, file architecture of a web server, Use of PHP Tags, Tag Styles, include, Processing GET and POST request, uploading files to server, cookies, sessions, Difference between MySQLi and PDO, CRUD - Create, Read, Update, Delete, records in database, pagination, login.

UNIT 2 [15 Hrs]

Introduction to Laravel, Routing in Laravel, MVC in Laravel, Caching in Laravel, Event subscribers in Laravel, Package Development, Templates, XML and AJAX, defining AJAX array, parse AJAX using JQuery, AJAX request, AJAX response.

UNIT 3 [15 Hrs]

Creating an Application, Database Configuration, Helpers in Laravel, Laravel Pagination, Laravel Security, Authentication Facade, Validation in Laravel, Eloquent ORM, Artisan Command Line Interface, Deploy Application using Laravel.

REFERENCE

Mandatory Reading:
1. Leon A, Zee S(2004), *Core PHP Programming* (3rd ed.).Prentice Hall Professional

2. Stauffer, M. (2019). *Laravel: Up & Running: A Framework for Building Modern PHP Apps.* O'Reilly Media.

Supplementary Reading:
1. Williams, H. E., & Lane, D. (2004). *Web Database Applications with PHP and MySQL: Building Effective Database-Driven Web Sites. O'Reilly Media, Inc.*

2. Brinzarea, B., & Hendrix, A. (2009). Ajax and PHP: Building modern Web

applications. Packt Publishing Ltd.

Web References:

1. https://www.w3schools.com/php/

2. https://www.tutorialspoint.com/php/index.htm

3. https://laravel.com/docs/6.x https://www.tutorialspoint.com/laravel/index.htm

Practical :Web Development with Laravel

Marks: 75

Duration: 45hrs

Credits: 03

1. 1. PHP Classes and instances, PHP Controls Structures [1P]

2. PHP Array Programming, Inheritance [1P]

4. CRUD using PHP database API's. [3P]

5. Fetch data from a form, validate and insert in the database, Delete data in the database ,Update data in the database, Display data from the database .Uploading files and session management. [1P]

6. Implementing MVC [2P]

7. Migrations in Laravel [1P]

8. Using Forms and Gathering Input in Laravel [1P]

9. Creating a registration & user login form in Larvael [1P]

10. Using Controllers and Routes for URLs and

APIs in Laravel [1P]

11. Eloquent ORM in Laravel [1P]

12. Creating and Using Composer Packages [1P]

13. Security & Session [1P]

SKILL ENHANCEMENT COURSE (SEC)

Course Title: Graphic Design
Course Code: COM-SEC4
Marks: 100
Credits: 4
Duration: 60 Hrs

Course Prerequisites: Nil

Course Objectives:

To understand the basic concept of Multimedia.
To develop skills for designing graphical images.
To develop skills for audio and video editing.
To acquire skills in using audio/video editing software.

Course Outcomes:

Upon completion of the course students will be able to:
CO1: Understand the concept of Multimedia – Team members and their roles.
CO2: Identify and describe the function of the general skill sets in the multimedia industry.
CO3: Classify and realize the types of Authoring tools and their functions.
CO4: Identify basic components of a multimedia project.
CO5: Analyze the requirements of Multimedia product.
CO6: Assemble and deliver multimedia projects

SYLLABUS:

UNIT I: Introduction to Multimedia and Computer Graphics: [15HRS]

Commonly used terms associated with multimedia like CDROM, Storyboard, Script and Authoring tools. Stages of a Multimedia Project-Planning and Costing, Designing and Producing, Testing and Delivering. The Multimedia team and their roles- Project Manager, Writer, Video specialist, Audio specialist and Multimedia programmer. Multimedia Software. Multimedia Hardware. Social & Ethical considerations, Digital Representations & Standards. A survey of Computer Graphic:
Computer Aided Design, Presentation Graphics, Computer art, Entertainment, Education and

Training, Visualization, Image Processing, Graphical User Interface

Multimedia Authoring Tools:
Types of authoring tools; Card and Page based tools; Icon-based; Time-based; Object-Oriented Tools.
TEXT: Text in graphics, character set, fonts, layout.
Character Attributes, Text Attributes, Marker Attributes, Bundled Text Attributes, Bundled Marker Attributes.

Vector graphics fundamentals, shapes, transforms and filters, Bitmapped graphics: resolution, image compression, manipulation, Geometrical transformations.
Activity: Design a Brochure for a given product, give details. Learn about different Image file Formats.

UNIT II: Audio, Video and Color [15HRS]

Basic Sound Concepts, Digitizing and processing sound, Music, Speech, Compression, formats, MIDI and Digital Audio Human vision, Camera systems, Gamma correction, Color matching, different Color models – RGB, CYMK, Transformations among color model.

Video:Analog and Digital Video, video standards, Video on PC. Introduction to graphics accelerator cards, Video Broadcast Standards - NTSC, PAL, SECAM, HDTV. Introduction to video capturing, Media & Instrument – Videodisk, DVCAM, Camcorder.

Recording Formats like S-VHA Video, Component (YUV), Component Digital, Composite Digital, and Video Hardware Resolutions.

Video Tips like shooting platforms, Lighting, Chroma Key or Blue Screen.

Activity:
1. Record and Edit the Sound file and Learn about Effects and Filters of sound
2. Prepare Video content with title and special effects.

UNIT III: Animation [15HRS] Basics of animation, Principle and use of animation in multimedia, Animation techniques-persistence of vision, animation file formats, Computer animation kinematics and Morphing. Design of Animation Sequence, General Computer Animation Functions, Computer Animation Languages, Key frame System, Particle systems – particle generation, modeling water, fire, explosions.

Effect of resolutions, pixel depth, Images size on quality and storage. Overview of 2-D and 3-D animation techniques and software. Animation on the Web – features and limitations, creating simple animations for the Web. Animation file

formats.
Activity: Animate 2D cartoon characters in real-time.

UNIT IV: Multimedia on the Web [15HRS]

Bandwidth relationship, broadband technologies, Text in the web – Dynamic and embedded font technology, Audio on the Web – Real Audio and MP3/MP4, Audio support in HTML, Graphics – HTML safe color palate, Interlaced V/s Non interlaced model, Graphics support in HTML, Image Map, Video on the Web – Streaming video.

Activity:

1. Prepare multimedia content for the web.

REFERENCES:

MANDATORY:

1. Chapman, N., & Chapman, J. (2005). Digital multimedia. John Wiley & Sons, Inc. 2. Li, Z. N., Drew, M. S., & Liu, J. (2004). Fundamentals of multimedia (pp. 253-265). Upper Saddle River (NJ): Pearson Prentice Hall.

3.Vaughan, Tay; Multimedia: Making it Work; Tata McGraw-Hill, 9th edition.

SUPPLEMENTARY:

1. Jeffcoate, J. (1995). Multimedia in practice. Technology and Applications. Great Britain: Prentice Hall.

WEB BASED:

1. https://www.gimp.org/
2. https://www.audacityteam.org/
3. https://swayam.gov.in/nd2_ugc19_hs42/
4. https://www.tutorialspoint.com/multimedia
5. https://libguides.bc.edu/

Course Title: Programming in Python
Course Code: COM-SEC5
Marks: 75
Credits: 3
Duration: 45 HRS

Course prerequisite: Nil

Course Objectives:
To provide skills of data analysis using Python

programming language.

Course Outcomes:

At the end of the course students should be able to :

CO1: Understand syntax of Python Programming.

CO2: Write program using conditional statements, loops.

CO3: Apply required List function.

CO4: Write Python program specific to the domain of the

given problem.

SYLLABUS:

UNIT I: Introduction to Python [15HRS] Motivation, programming paradigms, What Python can do, Python's technical strength, Python interpreter, Program execution, Execution model variations, How to run programs.

Basic Syntax

Variable Data Types, Operator, Conditional

Statements-if,if-else, Nested If else.Looping–For, While, Nested loops. Control Statements–Break, Continue, Pass.

UNIT II: String Manipulation, Tuple, Lists and Dictionaries:

[15HRS]

Accessing Strings,Basic Operations,String Slices,Function and Methods. Tuple and Lists

Introduction, Accessing list, Operations, Working with lists, Function and Methods. Introduction Accessing tuples, Operations, Working, Functions and Methods.

UNIT III: Functions, Modules and Input-Output [15HRS]

Introduction, Accessing values in dictionaries, Working with dictionaries, Properties, Functions**.** Functions. Defining a function, Calling a function, Types of functions, Function Arguments, Anonymous functions, Global and local variables.

Modules

Importing module. Math module. Random Module.

Exception Handling

Exception.ExceptionHandling-Exceptclause,Try,except,finallyclause.UserDefined Exceptions

Input-Output

Printing Onscreen,Reading data from Keyboard,Opening and closing file,Reading and writing files, Functions. Printing Onscreen,Reading data from Keyboard,Opening and closing file,Reading and writing files, Functions.

REFERENCES:

MANDATORY:

1. Mark Lutz, Learning Python, O'Reilly Media, Fifth Edition.

SUPPLEMENTARY:

1. Alex Martelli,(2006)Python–A Nutshell, O'Reilly Media, Second Edition. **2.** WesMc Kinney,(2012)Python for Data Analysis ,O'Reilly Media.

WEB BASED:

1. https://www.w3schools.com
2. https://www.tutorialspoint.com
3. https://www.javatpoint.com
4. https://www.geeksforgeeks.org
5. https://www.guru99.com

Practicals: Programming in Python
Credit: 01
Marks: 25
Duration:30 Hrs

List of Experiments using Python Language:

1. Write a python program to compute a given formula.
2. Write a python program to implement an if else statement.
3. Write a python program to implement nested if else statements.
4. Implement for and while loop in python. [2P]
5. Write a string manipulation in python. [2P]
6. Write a program to implement lists in python.
7. Write a python program to implement tuple.
8. Write a python program to implement a dictionary.
9. Write a python program to implement a function.
10. Implement Module in python
11. Write a python program to implement exception handling.
12. Write a python program to implement Input-Output File operations in python. [2P]

Paper Title : Web Development with FLASK
Paper Code : COM-SEC6
Marks : 75
Credits : 3
Hours:45Hrs

Course Prerequisites: Programming in Python, Database Management System

Course Objectives:
To develop skills to build dynamic web applications, ensuring security and scalability. Train students to deploy web applications and configure the same for a development and production environment.

Course Outcome:

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

CO1: Understand the basic of server client architecture on how request are made, handling those requests and providing an appropriate response.
CO2: Understand how to scale a web application.
CO3: Secure a web application from CSRF, Sql injection and XSS .
CO4: Handle dependencies management across multiple web application using python virtual environment.
CO5: Configure a web application for development and production.
CO6: Build a dynamic web site / web app.
CO7: Create API .
CO8: Deploying a web application

Syllabus:-

UNIT I : Introduction to Web Development: [15 HRS] Client-Server architecture, Requests: GET, POST. Web addresses and URL, Dynamic Websites, Static Websites, Database Servers and Web Servers. MVC and MVT.

Introduction to Flask. Flask and Other web frameworks. Introduction to python modules and pip. Python venv module, Installing modules using pip, dependency management.

Configuration and Deployment Flask configuration basic , Built in Configuration Values: SECRET_KEY, SESSION_COOKIE_PATH, LOGGER_NAME,

APPLICATION_ROOT Development / Production configuration, Configuration Best Practices. Deployment on Apache HTTP server.

Jinja 2.

Introduction to Jinja, Jinja Setup, Standard Context , Standard Filters , Controlling Auto escaping. Registering Filters, Context Processors.

UNIT II : Building a Minimal Application: [15 HRS]

Routing, Starting a flask server, URL building, Rendering Templates, Debugging, Accessing request data,HTML methods: GET POST, Cookies Sessions Handling. Redirecting and Errors. Message Flashing. Logging. Building a salable project structure.

FLASK WT Forms and FLASK SQL Alchemy (ORM):

WT Forrms Introduction, Creating Forms, Validating Forms , Securing Forms, File Uploads, CSRF protection. SQL Alchemy setup and installation, Creating database using Sql Alchemy, Simple Relationship, One to many and many to many relationship, Inserting records , Deleting records , Editing Records , Querying Records

UNIT III: Flask Security [15 HRS]

Session based authentication, Role management , Password hashing , Basic HTTP authentication Token based authentication ,Token based account activation ,Token based password recovery / resetting , User registration, Login tracking , JSON/Ajax Support

Flask Admin:

Getting Started Authorization & Permissions, Customizing Built-in Views Adding Your Own Views Working With the Built-in Templates .

REFERENCES:

MANDATORY:

1.Miquel Grinberg , "Flask Web Development ", O'REILLY

SUPPLEMENTARY:

1.Gareth Dwyer "Flask By Example ", Packt Publishing Limited

WEB BASED:

1. https://flask.palletsprojects.com/en/2.1.x/
2. https://dl.acm.org/doi/10.5555/2621997

3. https://www.fullstackpython.com/flask.html
4. https://coddyschool.com/upload/Flask_Web_Development_Developing.pdf 5. https://www.packtpub.com/product/mastering-flask-web-development-second edition/97 81788995405

Practicals: Web Development with FLASK
Credit: 01
Marks: 25
Duration:30 Hrs

1) Creating a basic application. (1P)
2) Working with the requests and jinja. (1P)
3) Form creating and validation with WT Forms. (1P) 4) Creating an admin panel with FLASK Admin. (1P) 5) Customizing FLASK admin panel. (1P)
6) Connecting database through SQL Alchemy. (1P) 7) Insert/Update/Delete/Query data using SQL Alchemy. (4P) 8) Securing your application with FLASK Security. (1P) 9) Creating a scalable project structure. (1P)
10) Build your own API (REST API). (2P)
11) Deploying and Configuring FLASK. (1P)