DEPARTMENT OF COMPUTER SCIENCE

MASTER OF COMPUTER APPLICATIONS

CURRICULUM AND SYLLABI

(effective from 2020 - 21)

MADRAS CHRISTIAN COLLEGE (AUTONOMOUS)

TAMBARAM, CHENNAI – 600 059

MADRAS CHRISTIAN COLLEGE (AUTONOMOUS)
DEPARTMENT OF COMPUTER SCIENCE

CURRICULUM FOR M.C.A. (EFFECTIVE FROM 2020 – 21)
PROGRAMME EDUCATIONAL OBJECTIVES (PEOs)

PEO 1: Offer wholesome education and build competent professionals who will possess strong fundamental knowledge, analytical skills and computational abilities to solve complex problems appropriate for global computing industry.

PEO 2: Create sustained learners to bring out novel ideas in addressing research issues and challenges.

PEO 3: Develop professional skills like teamwork, communication and management with professional ethics and values and upgrade their skills to become entrepreneurs.

PROGRAMME OUTCOMES (POs)

PO 1. **Problem analysis and design/development of solutions:** Ability to apply the knowledge of Computing, Mathematics and Science to provide solutions for complex problems, with appropriate consideration for public health and safety, legal, cultural, societal, and environment constraints.

PO 2. **Investigation of complex problems:** Use research-based knowledge and research methods to independently carry out research/investigation and synthesize information to provide valid conclusions.

PO 3. **Usage of emerging technologies and modern tools:** Ability to use emerging technologies, skills, resources and modern IT tools, with software engineering principles, to produce software products meeting the global standards.

PO 4. **Individual/team work:** Function effectively as an individual, and as a member or leader in diverse and multidisciplinary teams to accomplish targets with effective oral and written communication.

PO 5. **Professional Ethics:** Understand and commit to professional ethics and cyber regulations, responsibilities and norms of professional computing practices.

PO 6. **Lifelong Learning:** Recognize the need for learning with respect to the changes in technology and the ability to engage in independent and lifelong learning.

PROGRAMME OUTCOME – COURSE MATRIX

Semester	Course Name	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6
I	Data Structures and Algorithms	3	1		1		3
	Object-Oriented Programming	3	1		1		3
	Principles of Database Management Systems	3	2		2		3
	Elective - I						
	Elective - II						
	Data Structures and Algorithms Laboratory	2	1	1	1		2
	Programming Laboratory in C++	2	1	1	1		2
	RDBMS Laboratory	2	1		2		2
II	Artificial Intelligence	3	3	2	2		3
	.NET Technologies	3	1		2		3
	Web Programming	3	1		2		3
	Elective - III						
	Elective - IV						
	Advanced Java Programming Laboratory	3	1	2	2		2
	Programming Laboratory in .NET Technologies	2	1		2		2
	Web Programming Laboratory	2	1	2	2		2
	Soft Skill Programme						
III	Big Data Analytics	3	3	2	2		3
	Advanced Software Engineering	3	2	2	2	1	3
	Elective - V						
	Elective - VI						
	Elective - VII						
	Mobile Application Development Laboratory	2	1	2	2		2
	Data Analytics with Python Programming Laboratory	3	2	2	2		2
	Software Development Laboratory	3	2	2	3	2	2
	Soft Skill Programme						
IV	Project	3	2	2	2	2	3
	Internship	1	1	2	2	2	2

MADRAS CHRISTIAN COLLEGE (AUTONOMOUS)
DEPARTMENT OF COMPUTER SCIENCE
PROGRAM STRUCTURE FOR M.C.A. (EFFECTIVE FROM 2020 – 21)

	Course Name	**Hours / DO Cycle**	**Credits**	**ICA Marks**	**ESE Marks**	**Total Marks**
Semester I						
11	Data Structures and Algorithms	4	3	50	50	100
12	Object-Oriented Programming	4	3	50	50	100
13	Principles of Database Management Systems	4	3	50	50	100
14	Elective - I	4	3	50	50	100
15	Elective - II	4	3	50	50	100
16	Data Structures and Algorithms Laboratory	3	3	50	50	100
17	Programming Laboratory in C++	3	3	50	50	100
18	RDBMS Laboratory	4	3	50	50	100
	Total	30	24			
Semester II						
21	Artificial Intelligence	4	3	50	50	100
22	.NET Technologies	4	3	50	50	100
23	Web Programming	4	3	50	50	100
24	Elective - III	4	3	50	50	100
25	Elective - IV	4	3	50	50	100
26	Advanced Java Programming Laboratory	4	3	50	50	100
27	Programming Laboratory in .NET Technologies	3	3	50	50	100
28	Web Programming Laboratory	3	3	50	50	100
	Soft Skill Programme	2	4	50	NA	50
	Total	32	28			
Semester III						
31	Big Data Analytics	4	3	50	50	100
32	Advanced Software Engineering	4	3	50	50	100
33	Elective - V	4	3	50	50	100

34	Elective - VI	4	3	50	50	100
35	Elective - VII	4	3	50	50	100
36	Mobile Application Development Laboratory	4	3	50	50	100
37	Data Analytics with Python Programming Laboratory	3	3	50	50	100
38	Software Development Laboratory	3	3	50	50	100
	Soft Skill Programme	2	4	50	NA	50
	Total	32	28			
Semester IV						
41	Project (Duration : 4 months)		15	50	50	100
	Internship (Duration: 3 to 4 weeks)		2	-	-	-
	Total		17			
	Total		97			

ELECTIVES	
E1	Advanced Java Programming
E2	Block Chain and Cryptocurrency
E3	Cloud Computing
E4	Computer Networks and Security
E5	Content Management Systems
E6	Cryptography and Network Security
E7	Cyber Security and Forensics
E8	Data Warehousing and Mining
E9	Digital Image Processing
E10	Digital Marketing
E11	Ethical Hacking
E12	Free and Open Source Software
E13	Healthcare Information Management System
E14	Information Security
E15	Internet of Things

E16	Internet of Things and Cloud
E17	Machine Learning Techniques
E18	Mobile Application Development
E19	Mobile Commerce Technology
E20	Mobile Computing
E21	Natural Language Processing
E22	Object Oriented System Design
E23	Principles of Operating Systems
E24	Soft Computing
E25	Software Engineering, Project Management and Testing
E26	System Software
E27	User Interface and User Experience Design
E28	Directed Study

11 DATA STRUCTURES AND ALGORITHMS

Semester: I **Hours: 4** **Credits: 3**

OBJECTIVES:

- To impart knowledge about the concepts of data structures and algorithms.
- To train the students to design and analyze linear and non-linear data structures and algorithms.
- To make the students apply suitable data structures and algorithms for solving real-world applications.

Unit 1 - Introduction to Data Structures and Algorithms

Basic Data Types - Abstract Data Types; Structure, operations on them and implementation, Need for data structures. Fundamentals of Algorithmic problem solving, Analysis of Algorithm Efficiency - Orders of growth, Asymptotic and Empirical analysis of algorithms, Solving Recurrence Equations, Space-Time tradeoff. Arrays - Single and Multidimensional arrays, Insertion and deletion in arrays, Strings,

Unit 2 - Linear Data Structures

Stacks: Push, Pop, Array representation of a stack, Applications - Expression evaluation, Recursion – Tower of Hanoi. *Queues*: Enqueue, Dequeue, Array representation of a queue, Circular queues. *Lists:*Singly linked list, Memory representation of lists, Sorted and unsorted list, Doubly linked list, Circular lists, polynomial addition and subtraction using lists, Linked representation of stacks and queues

Unit 3 - Non-linear Data Structures

Trees: Basic Terminologies, Binary Trees and their memory representation, Tree traversals, Binary Search Trees – traversing, insertion and deletion. *Graphs*: Directed and Undirected graphs, Memory representation of graphs, Weighted graphs, Graph Traversal - BFS, DFS, Single-Source Shortest Path Problem - Dijkstra's algorithm, Minimum Cost Spanning Trees by Prim's and Kruskal's algorithm, All-Pairs Shortest Path Problem – Floyd-Warshall algorithm.

Unit 4 - Design and Analysis of Algorithms - I

Brute Force - Bubble sort, Sequential search, String search, Divide and Conquer - Merge sort, Quick sort, Decrease and Conquer: Decrease by one - Insertion sort, Decrease by a constant factor - Binary search, Analysis of all the techniques

Unit 5 - Design and Analysis of Algorithms - II

Transform and Conquer - Instance simplification-Presorting, Representation change-Heap construction, Dynamic programming–Minimum number of coins that make a given value, Greedy Technique - Knapsack problem, Backtracking - n-queens problem – Branch and Bound technique - Traveling Salesman Problem, Analysis of all the techniques.

OUTCOMES:

At the completion of the course, the learner will be able to

- Analyze the performance of algorithms using asymptotic notations.
- Evaluate and provide suitable techniques for solving a problem using basic properties of Data Structures.

- Illustrate different types of algorithmic approaches to problem solving and assess the tradeoffs involved.
- Understand the nature of problems and to develop prototypes or applications of varying complexities.

References

1. Kruse R.L, Leung B.P, Tondo C.L, *Data structures and Program design in C*, Second Edition, Pearson, 2007.
2. Mark Allen Weiss, *Data Structures and Algorithms in C*, Second Edition, Pearson, 2006
3. S.K.Basu, *Design Methods and Analysis of Algorithms*, Fourth edition, PHI, 2013
4. A.V.Aho, J.E. Hopcroft and J.D.Ullman, *The Design and Analysis of Computer Algorithms*, Pearson, 2002
5. AnanyLevitin, *Introduction to the Design and Analysis of Algorithm*, Third Edition Pearson Education Asia, 2016
6. Robert Sedgewick and Kevin Wayne, *Algorithms*, Fourth Edition, Pearson Education, 2012.

12 OBJECT-ORIENTED PROGRAMMING

Semester: I **Hours: 4** **Credits: 3**

OBJECTIVES:

- To understand and apply the OOP concepts using C++, an OOP specific programming language.
- To write programs to solve problems using the OOP language constructs.

Unit 1

OOP Paradigm: Comparison of programming paradigms - Key concepts of Object Oriented Programming - Abstraction, Encapsulation, information hiding, interface and implementation. **Functions** – Function overloading, default arguments, inline functions. Arrays, pointers, references and dynamic memory allocation.

Unit 2

Classes and Objects: Members, messages, constructors, destructors, access control specifiers, overloading member functions, overloading constructors, scope resolution operator. **Static members:** static member data, static member function, static class. **Operator overloading** - unary and binary operator overloading.

Unit 3

Friends: friend function, friend class. **Generic Programming:** Templates - Function templates - class templates – Standard Template Library. **Relationship among classes: Aggregation -** whole and part classes. **Using** – client class and server class

Unit 4

Relationship among classes: Inheritance: Reusability - Implementation of Inheritance - simple - multilevel - multiple – hybrid. Overriding member functions. **Polymorphism:**

Implementation of polymorphism - virtual functions - pure virtual functions, virtual base classes, abstract class.

Unit 5

I/O streams – stream input / output classes and objects, I/O Stream Manipulators, Files – File streams – sequential files – random access files. Exception handling: Exception – handling exceptions – Exception class.

OUTCOMES:

At the completion of the course, the learner will be able to

- Apply the concepts of data abstraction, encapsulation, overloading, inheritance and polymorphism and identify classes and objects for a given problem.
- Design and implement C++ programs for a given problem.
- Know how to use the OOP concept reusability to modify an existing program
- Know how to write generic programs using templates and STL

References

1. Paul Deitel and Harvey M. Deitel, *C++ How to Program*, Tenth Edition, Pearson India Education Services Pvt. Ltd, 2017
2. Bjarne Stroustrup, *The C++ Programming Language*, Fourth Edition, Addison-Wesley publishing company, 2013.
2. Herbert Schildt, *C++ From the Ground Up*, Third Edition, Osborne McGraw Hill, 2003.
3. Herbert Schildt, *C++ Complete Reference*, Fifth Edition, McGraw Hill Education, 2012.
4. Stephen Prata, *C++ Primer Plus*, Sixth Edition, Addison Wesley, 2013

13 PRINCIPLES OF DATABASE MANAGEMENT SYSTEMS

Semester: I **Hours: 4** **Credits: 3**

OBJECTIVES:

- To understand the fundamentals of data models and conceptualize and depict a database system using ER diagram.
- To make a study of relational database design.
- To learn in detail about query processing using SQL.
- To impart knowledge in transaction processing, concurrency control techniques and recovery procedures.
- To know about data storage techniques and Indexing.

Unit 1

Introduction: Purpose of Database System - Views of data - Data Models - Database System Architecture - Entity Relationship model: ER Diagrams - Relational Model Concepts - Relational Query Languages - Relational Algebra

Unit 2

Relational Database Design**:** Non-loss Decomposition - Functional Dependencies - 1NF - 2NF - 3NF - Dependency Preservation – Boyce-Codd Normal Form - Multi-valued Dependencies, 4NF - Join Dependencies, 5NF

Unit 3

SQL: Query Structure, Basic operations, Additional Operations, Set Operations, Null Values, Aggregate Functions, Nested Subqueries - Modification of the Database - Join Expressions, Views, Integrity Constraints, SQL Data Types, Authorization - Advanced SQL features - Accessing SQL from a Programming Language - Dynamic SQL (JDBC and ODBC), Embedded SQL, Functions and Procedural Constructs, Triggers.

Unit 4

Transactions**:** Concepts, Properties of Transactions, Serializability - Commit protocols: Two Phase Commit - Concurrency - Locking Protocols: Two Phase Locking - Deadlock - Recovery concepts, Log based recovery.

Unit 5

Overview of Physical Storage Media - RAID - Tertiary storage - File Organization - Hashing: Static and Dynamic Hashing - Indexing: Single level and Multi-level Indexes - Overview of special purpose databases

OUTCOMES:

At the completion of the course, the learner will be able to

- Understand the basic concepts of the database and data models.
- Design a database using ER diagrams and map ER into Relations and normalize the Relations.
- Develop a simple database application using normalization.
- Acquire the knowledge of query evaluation to monitor the performance of the DBMS.
- Acquire the knowledge about different special purpose databases and how to store it.

References

1. Abraham Silberschatz, Henry F. Korth, S. Sudharshan, *Database System Concepts*, Sixth Edition, McGraw Hill Education, 2010
2. C.J.Date, A.Kannan, S.Swamynathan, *An Introduction to Database Systems*, Eighth Edition, Pearson Education, 2006.
3. RamezElmasri, Shamkant B. Navathe, *Fundamentals of Database Systems*, Sixth Edition, Pearson Education Inc., 2011.
4. Raghu Ramakrishnan, *Database Management Systems*, Fourth Edition, Tata McGraw Hill, 2014.
5. G.K.Gupta, *Database Management Systems*, Tata McGraw Hill, 2011.

16 DATA STRUCTURES AND ALGORITHMS LABORATORY

Semester: I **Hours: 3** **Credits: 3**

OBJECTIVES:

This Laboratory aims at making students to write algorithms and convert them to C programs for various problems using:

- linear and non-linear data structure
- searching, sorting and backtracking techniques

Concepts to be covered:

1. Stacks

- Array implementation
- Linked list implementation
- Evaluation of expression

2. Queues
   - Array implementation
   - Linked list implementation
3. Linked List
   - Singly linked list (Traversing, Insertion, deletion)
   - Doubly linked list (Traversing, Insertion, deletion)
   - Circular linked list (Traversing, Insertion, deletion)
4. Sort
   - Insertion sort
   - Selection sort
   - Merge sort
   - Quick sort
5. Search
   - Linear search
   - Binary search
6. Trees
   - Binary Tree Traversals (Preorder, Inorder, Postorder)
   - BST insertion and deletion
7. Graphs
   - Traversal (Breadth first search, Depth first search)
8. Graph Applications
   - Single source shortest path (Minimum Spanning Tree) - Dijkstra's and Prim's algorithm
   - All pairs shortest path – Warshall's algorithm / Floyd's algorithm
9. n-Queens algorithm using Backtracking

OUTCOMES:

Write and execute C programs on,

- Linear Data Structures like arrays, stacks, queues and linked lists.
- Non-linear Data Structures like binary search trees and graphs.
- Searching and Sorting techniques
- Backtracking

17 PROGRAMMING LABORATORY IN C++

Semester: I **Hours: 3** **Credits: 3**

OBJECTIVES:

- To learn the features in C++ without objects
- To learn Object-based features in C++
- To learn compile time polymorphism in C++
- To use inheritance for reusability of code
- To learn generic programming

1. **Features in C++ not involving objects**
    - Function overloading
    - Default argument
    - Reference variable
    - Inline function
    - Template function
    - Command line arguments

2. **Object-based Programming features in C++ - Class & Object**
    - Class, object, member functions, messages, constructors, destructor & scope resolution operator
    - Operations using two objects (object 1 receiver of message, object 2 parameter)
    - Operations using three objects (object 1 receiver of message, object 2 parameter, object 3 returned object)
    - Aggreagation : Whole and part class

3. **Object-based Programming features in C++- Advanced**
    - Operator Overloading (Unary and & Binary operators)
    - Templates (Template class)
    - Friends (Friend Function and Friend Class)

4. **Object-Oriented Programming Features in C++ - Inheritance**
    - Single Inheritance (Public)
    - Multi-level Inheritance (Public, Protected and Private)
    - Multiple Inheritance
    - Multiple Inheritance-Diamond Lattice Problem(Using Virtual Base class)

5. **Runtime Polymorphism**: (Inheritance, dynamic binding, virtual function)

6. **Abstract class** (Inheritance, pure virtual function)

7. **Static**
    - Static data
    - Static member function
    - Static class

8. **Streams**
    - Input output streams:
    - File Streams

9. **Exception Handling**

10. **Standard Template Library**

OUTCOMES:

At the completion of the course, the learner will be able to

- Write programs in C++ to implement non object-based features such as default arguments, function overloading, inline function, function templates
- Write programs using classes, objects
- Write programs that reuse other programs

- Implement polymorphism
- Identify and implement generic programs

18 RDBMS LABORATORY

Semester: I **Hours: 4** **Credits: 3**

OBJECTIVES:
- To understand the concepts of DBMS.
- To familiarize with SQL queries.
- To write stored procedures in DBMS.
- To learn front end tools to integrate with databases.

Backend SQL Server:
Connecting the front end and the back end.

Front-end VB.NET:
Reports- GridView.

Advanced SQL commands using more than one table

- **DDL statements**
  Create Table, Drop and Alter Table

- **DML statements**
  Insert, Select ,Update, Delete, Sub queries, Case Expression, Sql Predicates, Logical Connectives, Aggregate Functions, Set Operations

- **TCL Statements**
  Commit, Rollback, Savepoint.

- **Programmatic SQL**
  Triggers

OUTCOMES:
At the completion of the course, the learner will be able to
- Design and Implement databases
- Formulate complex queries using SQL
- Design and Implement applications that have GUI and access databases for backend connectivity

References

1. Petkovic, *Microsoft SQL Server 2008 A Beginner's Guide*, Tata McGraw Hill Edition, 2008
2. Itzik Ben-Gan, Microsoft SQL Server 2008 T-Sql Fundamentals, Microsoft Press, Indian Reprint 2009.
3. Paul Nielsen, *Microsoft SQL Server 2008 Bible*, Wiley India, Reprint 2010.
4. Bill Sheldon, *Professional Visual Basic 2010 and .NET 4*, Wrox, Reprint 2010.

21 ARTIFICIAL INTELLIGENCE

Semester: II **Hours: 4** **Credits: 3**

OBJECTIVES:

- To understand the fundamentals of computational intelligence
- To understand the various characteristics of Intelligent agents
- To learn about the different search strategies in AI
- To know about the various knowledge representation methods in solving AI problems
- To understand the features of neural network and its implementation
- To know about the various applications of AI.

Unit 1 Artificial Intelligence and Intelligent Agents

Introduction: Approaches to AI: Thinking Humanly, Thinking Rationally, Acting Humanly, Acting Rationally. Foundations of AI – Applications of AI. Intelligent agents: Agents and environments – Rational agents – Rationality – Nature of environments: Specifying the PEAS description of the task environment, Properties of task environment. Structure of agents – Agent programs – Basic types of Agent programs: Simple reflex agents, Model-based reflex agents, Goal-based agents, Utility-based agents. Learning Agents.

Unit 2 Problem-solving and Search

Problem-solving agents: Well-defined problems and solutions, Formulating problems. Searching for solutions - Uninformed search strategies: Breadth-first search, Depth-first search, Bidirectional search. Informed search strategies: Greedy best-first search, Recursive best-first search. Online search agents. Adversarial search: Games - Optimal Decisions in Games – Alpha-Beta Pruning.

Unit 3 Knowledge, Reasoning and Planning

Knowledge-based agents - Propositional logic - Agents based on propositional logic - propositional inference. First order (Predicate) logic – Inference in First order logic. Expert System. Knowledge representation: Categories and objects – Reasoning systems for categories. Classical Planning – Algorithms for planning as State-Space Search. Planning and acting in the real world.

Unit 4 Learning

Learning from Examples: Forms of Learning – Supervised Learning – Learning Decision Trees – Artificial Neural Networks. Knowledge in Learning: A Logical Formulation of Learning – Explanation-Based Learning – Inductive Logic Programming. Learning with hidden variables – Unsupervised learning – Clustering. Reinforcement learning – Passive Reinforcement Learning – Active Reinforcement Learning.

Unit 5 AI Applications

Text Classification - Information Retrieval – Information Extraction - Machine Translation – Speech Recognition – Object Recognition – Robotics

OUTCOMES:

At the completion of the course, the learner will be able to

- Implement computational intelligence through applications
- Use appropriate search algorithms for any AI problem
- Understand knowledge representation methods and apply approximate reasoning

- Represent a problem using first order and predicate logic
- Apply reinforcement learning techniques for finding a solution to a problem

References

1. Stuart J. Russell and Peter Norvig, *Artificial Intelligence A Modern Approach*, Third Edition, Pearson Education, Inc., 2010
2. Elaine Rich, Kevin Knight, Shivashankar B. Nair, *Artificial Intelligence*, Third Edition, Tata McGraw-Hill Publishing Company Limited, 2009.
3. David Pool, Alan Mackworth, *Artificial Intelligence: Foundations of Computational agents*, Cambridge University, 2011.
4. Christopher M. Bishop, *Pattern Recognition and Machine Learning*, Springer, 2013.
5. Nils J.Nilsson, *The Quest for Artificial Intelligence: A History of Ideas and achievements*, Cambridge University Press, 2010.

22 .NET TECHNOLOGIES

Semester: II **Hours: 4** **Credits: 3**

OBJECTIVES:

- To understand .NET Platform and its core functionalities.
- To develop windows and web applications with Microsoft SQL and Visual Studio.
- To understand and develop user defined Applications using MVC framework.
- To strengthen Object Oriented Programming using advance C# concepts.

Unit 1

Introducing Microsoft .NET: Microsoft .NET platform: .NET Enterprise Servers, .NET framework and .NET Building block Services - .NET Namespaces.Common Type System(CTS), Common Language Specification(CLS) and CLR Execution (Class loader, verifier, JIT compilers).

Unit 2

Windows Forms, ASP.NET and ADO.NET : Windows Forms: Windows Forms.NET classes, Windows controls, Data binding, Developing a windows application using C#. Web Forms: ASP.NET over ASP, Main classes in ASP.NET, Web Controls, Web Forms Syntax, Components and life cycle, Developing a simple ASP.NET page. Handling data: Benefits of ADO.NET, ADO.NET Architecture, Main classes in ADO.NET, Developing a Windows/Web application using database

Unit 3

ASP.NET MVC: Overview, Benefits, MVC Pattern, Life Cycle, Controllers, Actions, Views, Data Model. Model Binding, using Databases

Unit 4

C# Classes, Objects and Object Oriented Features: Type System, Boxing and Unboxing, Access control specifiers, Arrays, Iterators, Structs, Static -static data, static members and static classes, Object-Oriented Features-Class Inheritance, Polymorphism and Interfaces.

Unit 5

Advanced C# : Properties, Indexers , Exception handling, Namespaces, Delegates and event handlers, Attributes, Overloading Operators-unary and binary operator overloading, Conversion operators, Nested Types, Anonymous Methods, Generic Collections.

OUTCOMES:
At the completion of the course, the learner will be able to

- To design Websites using MVC and .NET framework.
- Gain more knowledge in advance C# Concepts.

References

1. Thuan Thai, *.NET Framework,* Third Edition, O'Reily publications, 2009
2. David S Platt, *Introducing Microsoft .NET* , Third edition, Microsoft press, 2003
3. Adam Freeman, *Pro ASP.NET MVC 5 Client*, Fifth Edition, Aprèss, 2013
4. Herbert Schildt, *C# 4.0 The Complete Reference*, Third Edition Tata McGraw Hill, 2010
5. Deitel & Deitel, *C# 2012 for Programmers*, Fifth Edition, Pearson, 2010
6. Tom Archer, *Inside C#*, Fourth Edition, Microsoft corporation, 2001
7. Dotnet_core, tutorialspoint.com/dotnet_core/index.htm
8. https://www.tutorialspoint.com/asp.net_mvc/index.htm

23 WEB PROGRAMMING

Semester: II **Hours: 4** **Credits: 3**

OBJECTIVES:

- To understand the concepts and architecture of the World Wide Web.
- To understand and practice markup languages
- To understand and practice embedded dynamic scripting on client side Internet Programming
- To understand and practice web development techniques on client-side.

Unit 1

HTML5 & JAVA SCRIPT: Advanced HTML5 tags - Form Controls - Embedding JavaScript in HTML, Dialog boxes - alert, confirm and prompt methods - Form processing: Form Validation and Event handling.

Unit 2

Angular JS: Basics of AngularJS – MVC – Filters and Modules – Directives – Working with Forms – Creating and Using Services

Unit 3

PHP Basics: String Manipulation – Creating, Accessing, Searching Replacing and Formatting strings; Control Structures – Decisions and Loops; Functions – Built-in and User Defined Functions, Passing parameters and Returning values.

Unit 4

Advanced PHP: Handling HTML Forms: Capturing Form Data - Regular Expressions: Pattern matching, Multiple matches and Replacing text - Classes And Objects - Cookies and Session.

Unit 5

PHP and MySQL: MySQL Database: Making a Connection – Handling Errors - Creating Tables – Insert, Update and Delete Records with PHP - Retrieving data using Select.

OUTCOMES:

At the completion of the course, the learner will be able to

- Create a basic website using HTML.
- Design and implement dynamic web page with validation using JavaScript objects and by applying different event handling mechanisms.
- Design rich client presentation using Angular JS.
- Design and implement simple web page in PHP.
- Design front end web page and connect to the back end databases

References

1. Peter Lubbers, Brian Albers, Frank Salim, *Pro HTML5 Programming*, APRESS, 2010
2. Pawel Kozlowski and Peter Bacon Darwin, *Mastering Web Application Development with AngularJS*, Packt Publishing, 2013.
3. Thomas A.Powell and Fritz Schneider, *JavaScript: The Complete Reference*, Tata McGraw Hill, 2002.
4. James Jaworski, *Mastering JavaScript*, First Edition, BPB Publications, 1999
5. Steceb Holzner, *PHP*: *The Complete Reference,* Tata McGraw Hill, 2007
6. Bob Boiko, *Content Management Bible*, 2nd Edition, Wiley, 2004.

26 ADVANCED JAVA PROGRAMMING LABORATORY

Semester: II **Hours: 4** **Credits: 3**

OBJECTIVES :

- To install and configure Eclipse IDE and Tomcat Server.
- Learn to access the MYSQL database using JDBC.
- To create dynamic web applications using Servlets and JSP.
- Understand and learn to map Java classes and object associations with relational database tables using Hibernate.
- To create MVC web applications using Struts frameworks.

1. **Runtime Environment**
   - Setting up the Runtime Environment for creating and deploying Servlet and JSP using Eclipse IDE and Tomcat Server.
2. **Servlet**
   - Basic Servlet Programming
   - HTML to Servlet Applications
   - Servlet Collaboration-Request Dispatcher
   - Session Management and Implementation of Cookies using Servlet
   - Developing a web application with MySQL Database using Servlet
3. **JDBC**
   - Implementation of any Information System using JDBC.

4. **JSP**
    - Basic JSP Programming- JSP Scripting elements, Directives and Actions.
    - Designing web applications with JSP – Form Processing and Database access.

5. **Hibernate Framework**
    - Hibernate Query Language
    - Java application with Database access using Hibernate
6. **Struts Framework**
    - Interceptors
    - ValueStack
    - ActionContext and ActionInvocation
    - Object Graph Navigation Language **(**OGNL)
    - Struts Validation
    - MVC based web application development using Struts Framework

OUTCOMES:

At the completion of the course, the learner will be able to

- Develop dynamic web applications using Servlet and JSP.
- Manage Sessions using Servlet Cookies.
- Learn MYSQL database connections using JDBC.
- Implement Hibernate framework by mapping Java application with database.
- Create MVC based web applications using Struts.

References

1. www.javatpoint.com/jsp-tutorial
2. www.javatpoint.com/servlet-tutorial
3. www.tutorialspoint.com/hibernate/hibernate_query_language.htm
4. www.javatpoint.com/hibernate-tutorial
5. www.javatpoint.com/struts-2-tutorial
6. www.tutorialspoint.com/struts_2/struts_examples.htm

27 PROGRAMMING LABORATORY IN .NET TECHNOLOGIES

Semester: II **Hours: 3** **Credits: 3**

OBJECTIVES:

- To develop windows and web applications in MVC .Net platform.
- To strengthen Object Oriented Programming using advance concepts in C#.

C#:

1. Simple Computations
2. Classes and methods
3. Constructors with parameters
4. Pass by values and pass by reference
5. Arrays
6. Structures
7. Enumerator
8. Jagged arrays

9. Method Overloading
10. Static Members
11. Operator Overloading
12. Inheritance
13. Virtual Methods
14. Abstract Class
15. Indexers
16. Delegates and Events
17. Interface
18. Exception Handling
19. Generic Method

.NET:

1. Windows Application with Database Connectivity using C# and ADO.NET
2. Web Application with Database Connectivity using C#, ASP.NET and ADO.NET
3. Web Application with Database Connectivity using ASP.NET MVC
4. Using ILDASM.exe
5. Creating and using strong Assemblies

OUTCOMES:

At the completion of the course, the learner will be able to

- To design Websites using .NET framework.
- Gain more knowledge in advance C# Concepts.

28 WEB PROGRAMMING LABORATORY

Semester: II **Hours: 3** **Credits: 3**

OBJECTIVES:

- Try and develop the most important technologies that are being used today by web developers to build a wide variety of web applications.
- To build web applications using proven developer tools and message formats.
- To understand and practice web development techniques on client-side
- Web applications using technologies such as HTML, Javascript, Angular JS and PHP.

JAVA SCRIPT using HTML controls

- Embedding JavaScript in HTML.
- Variables, operators, statements, functions.
- Dialog box - alert, confirm and prompt methods.
- Form processing: form validation.
- Event handling: link, document, image, form, mouse and keyboard events.

Angular JS: Basics of AngularJS – MVC – Filters and Modules – Directives – Working with Forms – Creating and Using Services

PHP:

String Manipulation, Control Structures, Functions, Working With Forms, Working With Regular Expressions, Introduction To Database, Advance PHP techniques.

MySQL: MySQL Database: Making a Connection – Handling Errors - Creating Tables – Insert, Update and Delete Records with PHP - Retrieving data using Select.

OUTCOMES:

- Develop simple web applications using scripting languages.
- Implement server side and client side programming develop web applications with various web technology concepts.
- Design a Web application using various PHP techniques.
- Develop an application using HTML5, AngularJS & PHP and store data by connecting using MySQL.

References

1. Thomas A.Powell and Fritz Schneider, *JavaScript: The Complete Reference*, Tata McGraw Hill, 2002.
2. James Jaworski, *Mastering JavaScript*, First Edition, BPB Publications, 1999
3. Steceb Holzner *PHP*: *The Complete Reference*, Tata McGraw Hill, 2007
4. Steven Holzner, *Teach Yourself XML in 21 Days*, Third Edition, Pearson Education, 2004.
5. Russell Nakano, *Web Content Management: A Collaborative Approach*, Addison-Wesley Professional, 2002.
6. Vikram Vaswani, *Mysql: The Complete Reference*, Tata McGraw Hill, 2017.

31 BIG DATA ANALYTICS

Semester: III **Hours: 4** **Credits: 3**

OBJECTIVES:

- To give an exposure on the basic concepts, methods, technology and tools of data analytics and make them understand the competitive advantages of big data analytics.
- To make the students understand the different characteristics and requirements of big data frameworks.
- To explain the concepts of distributed file system and Map Reduce programming.
- To make them learn and implement data analysis like correlation, regression, t-test, analysis of variance and time-series analysis
- To impart knowledge on Hadoop related tools for big data analytics
- To explain storage and processing of large data using NoSQL database

Unit 1 - Introduction to Big Data

Characteristics of Big Data and its importance – Applications of Big data - Four Vs of Big Data - distributed file system - Web data - Challenges of conventional systems, Drivers for Big data - Evolution of Analytic scalability - analytic processes and tools - Modern data analytic tools. Algorithms using map reduce, Matrix-Vector Multiplication by Map Reduce.

Unit 2 - Introduction to Data Analytics

Descriptive Statistics – Frequency and Contingency tables – Correlations – t-tests – Nonparametric tests of group differences. Regression: OLS regression, regression model. *Analysis of Variance:* Fitting ANOVA models, one-way ANOVA, one-way ANCOVA, two-way factorial ANOVA, repeated measures ANOVA, MANOVA, ANOVA as regression.

Text Analytics: Techniques, Information extraction, Tools, Applications. Case studies - correlation and regression study, t-test, ANOVA and text analysis table using Python

Unit 3 – Power and Time Series Analysis

Power Analysis : Hypothesis testing, implementing power analysis, creating power analysis plots. *Time Series Analysis:* Introduction to time series analysis, Forecasts using Exponential Smoothing, ARIMA Models. Case study of time series analysis using Spark Programming in Python

Unit 4 - Introduction to Hadoop

The Hadoop Distributed File System – Components of Hadoop- Analyzing the Data with Hadoop- Scaling Out- Hadoop Streaming- Design of HDFS-Java interfaces to HDFS Basics. Developing a Map Reduce Application. Job Scheduling-Shuffle and Sort – Task execution. Hadoop YARN

Unit 5 -NoSQL Data Management

NoSQL Databases : Schema-less Models: Increasing Flexibility for Data Manipulation-Key Value Stores- Document Stores – Tabular Stores – Object Data Stores – Graph Databases Hive – Sharding –Hbase

OUTCOMES:

- Understand the big data system and the need of analytical process.
- Differentiate between conventional and contemporary distributed framework and characterize storage and processing of large data.
- Perform data analysis using appropriate statistical methods and tools.
- Decompose a problem into map and reduce operations for implementation.
- Design programs to analyze large scale text data.
- Understand how to leverage the insights from big data analytics
- Implement and demonstrate the use of the hadoop eco-system.
- Perform analytics on real-time streaming data
- Understand the various NoSql alternative database models

References

1. Tom White, *Hadoop: The Definitive Guide*, Fourth Edition, O'reilly Media, 2015.
2. Chris Eaton, Dirk DeRoos, Tom Deutsch, George Lapis, Paul Zikopoulos, *Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data*, McGrawHill Publishing, 2012
3. Anand Rajaraman and Jeffrey David Ullman, *Mining of Massive Datasets*, Cambridge University Press, 2012
4. Glenn J. Myatt, *Making Sense of Data*, Wiley, 2007
5. Pete Warden, *Big Data Glossary*, O'Reilly, 2011
6. Michael Minelli, Michele Chambers, Ambiga Dhiraj, *Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Businesses*,Wiley Publications, 2013
7. Avril Coghlan, *A Little Book of R For Time Series*, Release 0.2, Parasite Genomics Group, 2018
8. Robert I., Kabacoft, *R in Action: Data Analysis and Graphics with R*, Manning Publication Co., 2011.

9. David Loshin, *Big Data Analytics: From Strategic Planning to Enterprise Integration with Tools, Techniques, NoSQL, and Graph*, Morgan Kaufmann/Elsevier Publishers, 2013.
10. https://www.tutorialspoint.com/hadoop/index.htm
11. https://hadoop.apache.org/
12. https://www.tutorialspoint.com/pyspark/index.htm

32 ADVANCED SOFTWARE ENGINEERING

Semester: III **Hours: 4** **Credits: 3**

OBJECTIVES:

- To present the contemporary industry relevant Software engineering practices
- To improve the skillset of students for better employability

Unit 1
Overview of Traditional SE - **Agile** Development Methods: Overview, Agile Manifesto, Refactoring Techniques, Limitations of Agile Process - Agile Modeling: Principles, Comparing XP and Agile Modeling - Lean Software Development

Unit 2
Scrum Methodology- The roles of Scrum, Advantages of Scrum - Dynamic Systems Development Methodology: Overview, Principles of **DSDM**, Phases of DSDM, Core Techniques

Unit 3
Extreme Programming (XP)- Introduction, XP Practices: Coding Practices, Developer Practices, Business Practices - XP Events- Iteration Planning- Stories and tasks, Estimates and schedules, First iteration, Iteration, Releasing Roles in XP: Customer's Roles, Developer's Roles, Supplementary Roles - Coding XP Style

Unit 4
XP Tools: **DevOps**: Need, Working of Devops - DevOps Tools – Benefits - DevOps Benefits - DevOps Use Case - Roles and Responsibilities of a DevOps Engineer

Unit 5
AI-Driven Software Engineering: Traditional software development Vs Machine learning model development, Design, Testing, Automatic code generation; AI Software Development Tools

OUTCOMES:
At the completion of the course, the learner will be able to

- Acquire knowledge on current software development models
- Understand the difference between traditional and current methods of software development
- Design, develop and deploy applications using industry relevant methods to software development
- Build and test applications with speed and quality

- Understand the evolution of software engineering for AI applications
- Conceptualize and design solutions to real-life problems using AI-enabled software engineering methods

References

1. Roger Pressman and Bruce Maxim, *Software Engineering: A Practitioner's Approach*, Ninth Edition, 2020
2. Robert C. Martin, *Agile Software Development: Principles*, Patters and Practices
3. Kenneth S. Rubin, *Essential Scrum: A Practical Guide to the Most Popular Agile Process*, First Edition, 2012.

36 MOBILE APPLICATION DEVELOPMENT LABORATORY

Semester: III **Hours: 5** **Credits: 3**

OBJECTIVES :

- To install and configure an Android application runtime environment using Eclipse IDE.
- Learn how to develop Android applications using Android Widgets.
- To develop Gesture based UI, Multimedia, SMS, Phone Call, Sensor and Location Based Android application.
- To implement Android database applications using SQLite.

1. Runtime Environment
   - Android Development Environment- Eclipse IDE and Emulator
   - Building a simple Android application using Eclipse

2. User Interface Design
   - Applications using GUI components- Activities
   - Styles and Themes
   - Fragments
   - Linking Activities using Intents
   - Layout Managers
   - Views- ListView, Spinner View, ImageView and GridView
   - Event Handling
   - Gesture Based Application

3. Multimedia application

4. Graphical and Animation

5. Telephony
   - SMS
   - Phone Call

6. Notification and Alarm

7. JSON Communication

8. Location based applications-GPS

9. Data retrieval applications using SQLite

10. Gaming application

11. Sensor Application

OUTCOMES:

At the completion of the course, the learner will be able to

- Learn to set up an Android development environment.
- Develop Android applications such as Gesture based UI, Multimedia, SMS, Phone Call, Sensor and Location Based application using Android UI toolkits and frameworks.
- Learn to connect Android Application with SQLite database.

References

1. www.tutorialspoint.com/android/index.
2. www.javatpoint.com/android-tutorial
3. www.vogella.com/articles/Android/article.html

37 DATA ANALYTICS WITH PYTHON PROGRAMMING LABORATORY

Semester: III **Hours: 4** **Credits: 3**

OBJECTIVES:

- To learn Python Environment set up.
- To analyse data using Python.
- To work with different types of data.
- To prepare data for analysis.
- To understand the importance of Data Visualization to help make more effective business decisions.
- To perform simple Statistical analysis.
- To solve analytical problems in real-world scenarios.

1. Establishing a Python environment for Data Analysis using ANACONDA **NAVIGATOR**

2. Essential Python Packages for Data Analysis
   - NumPy, Matplotlib and Pandas
   - Scikit-learn, StatsModels, SciPy and Seaborn

3. Importing and exporting datasets in Python

4. Data Processing
   - Data Operations
   - Data Cleansing-Identify and Handle Missing Values
   - Data Formatting
   - Data Normalization Sets

- Binning
- Indicator variables

5.Data Frame Manipulation using Pandas
- Descriptive Statistics
- Basic of Grouping
- ANOVA
- Correlation

6. Data Visualization using Matplotlib and Seaborn
- Basic plotting with Matplotlib
- Line plots, Area plots, Histograms, Bar charts, Box plots and Pie charts
- Statistical graphics using Seaborn

7. Building Machine Learning Regression models using Scikit-Learn
- Simple and Multiple Linear Regression

8. Building Data Pipelines
- Polynomial Regression and Pipelines
- R-squared and MSE for In-Sample Evaluation
- Prediction and Decision Making

9. Model Evaluation
- Model Evaluation
- Over-fitting, Under-fitting and Model Selection
- Ridge Regression
- Grid Search
- Model Refinement

OUTCOMES:

At the completion of the course, the learner will be able to

- Learn to work with data in Python for reading and writing files, loading, working, and saving data.
- Learn how to interpret data in Python using multi-dimensional arrays in NumPy, manipulate DataFrames in Pandas, use SciPy library of mathematical routines and execute machine learning using Scikit-Learn.
- Learn to perform data analytics using Python libraries.
- Gain knowledge on several data visualization libraries in Python, including Matplotlib and Seaborn.
- Learn to build Machine Learning Regression models.
- Develop creative and effective solutions to solve real time business problems.

References

1. Fabio Nelli, *Python Data Analytics with Pandas, Numpy, and Matplotlib*, Second Edition, Apress, 2018.
2. William McKinney, *Python for Data Analysis: Data Wrangling with Pandas, NumPy, and Ipython*, Second Edition, O'Reilly, 2017.

3. Fabio Nelli, *Python Data Analytics: Data Analysis and Science Using Pandas, matplotlib, and the Python Language*, APress, 2015.
4. Ken Black, *Business Statistics for Contemporary Decision Making*, Sixth Edition , Wiley, 2009
5. www.tutorialspoint.com/python_data_science/index.htm

38 SOFTWARE DEVELOPMENT LABORATORY

Semester: III **Hours: 4** **Credits: 3**

OBJECTIVES:

This Laboratory aims at developing a mini-project by students working in a team and doing the following activities:

- Each student develops few modules in the mini-project
- Integrate all the modules into a single project.
- Follow the project life cycle with requirement analysis, specification, design, coding, implementation, testing and documentation

Areas

- Android using Java or Kotlin
- Cryptography using C# or Java
- Data Analytics using Python
- Digital Image Processing using Java or Python
- Natural Language Processing using Java or VB.NET
- Networking using Java
- Web applications using PHP MVC Framework like Laravel or CodeIgniter
- Web applications using Angular, Node JS/ React JS
- Web applications using Java Frameworks like Hibernate, Struts or Spring
- Web applications using MVC, C# and .NET
- Machine Learning and Deep Learning using Python

OUTCOMES:

- Develop a mini-project by a team using any one of the cutting-edge technologies.
- Exercise team work in developing and integrating into a single project.

E1 ADVANCED JAVA PROGRAMMING

Elective **Hours: 4** **Credits: 3**

OBJECTIVES:

- Learn to create and deploy Servlets and JSP using Tomcat Server.
- Learn to develop Dynamic web applications using Servlet and JSP with MYSQL database connectivity.
- Learn how to manage sessions using Servlet Cookies.
- Understand the fundamentals of JDBC.
- To design Java applications with Database access using Hibernate.

- Learn to develop MVC based web application development using Struts Framework.

Unit 1

Web Component- Servlet: Need for dynamic content, Servlet- overview, Servlet API and Lifecycle –using Tomcat – Start Tomcat -Start Browser and request Servlet – Servlet API – Handling HTTP requests and responses-Servlet Collaboration-Request Dispatcher, Session tracking – Cookies; Creating a web application , Compiling and building the web application

Unit 2

JDBC and JSP: JDBC Introduction- JDBC driver types - Connections using JDBC – Declarative approach – Programmatic approach – Connecting to Data Source – Data Manipulation; **Java Server Pages**: JSP Introduction -JSP v/s Servlets, Lifecycle of JSP, Comments, JSP scripting elements, Directives and Action - Implicit objects- Working with variables scopes-Exception -JSP tag libraries (JSTL)-JSP Form Processing with Database access.

Unit 3

Hibernate: Object Relational Mapping (ORM), Java ORM Frameworks**,** Hibernate Introduction , Databases and Technologies supported, Setting up Hibernate Environment, Architecture – Hibernate Mapping -Understanding O/R mapping –Hibernate Query language (HQL): HQL clauses: SELECT, FROM, WHERE, AS, ORDER BY, GROUP BY, INSERT, UPDATE, DELETE.

Unit 4

Advanced Frameworks: Advanced Frameworks - MVC framework - **Understanding Struts**: Introduction, Struts Features, Control Flow, Installing and setting up struts, Getting started with Struts, Struts framework core components-Interceptors, ValueStack, ActionContext, ActionInvocation, Object Graph Navigation Language **(**OGNL).

Unit 5

Struts: Struts Architecture, Struts Action (POJO), Struts Configuration File-Validation-Custom Validation, Bundled Validators and Ajax Validation., Building model view controller component -Case studies – Current trends.

OUTCOMES:

At the completion of the course, the learner will be able to

- Work with Servlet and JSP to develop dynamic web applications.
- Learn database connections using JDBC.
- Understand the importance of advanced frameworks such as Hibernate and Struts.

References

1. Herbert Schildt , *Java The Complete Reference*, Eleventh Edition, McGraw Hill, 2018.
2. Paul Deitel and Harvey Deitel, *Java How to Program*, Ninth Edition, Prentice Hal, 2012.
3. Budi Kurniawan, *Servlet & JSP: A Tutorial*, Second Edition, Brainy Software, 2015.
4. Byyan Basham, Kathy Sierra, Bert Bates, *Head First Servlets and JSP*, Second Edition, O'Reilly Media Incorporated, 2008.
5. Larne Pekowsky, *JavaServer Pages*, Second Edition, Pearson Education, 2008.
6. www.javatpoint.com/jsp-tutorial

7. www.javatpoint.com/servlet-tutorial
8. www.tutorialspoint.com/hibernate/hibernate_query_language.htm
9. www.tutorialspoint.com/struts_2/struts_examples.htm

E2 BLOCKCHAIN AND CRYPTOCURRENCY

Elective **Hours: 4** **Credits: 3**

OBJECTIVES:

- To understand the need of Block chain applications.
- To understand the usage of Bitcoins as decentralized cryptocurrency.
- To know about the emerging trends of Cryptocurrency as a Digital Asset.

Unit 1

Basics: Distributed Database, Two General Problem, Byzantine General problem and Fault Tolerance, Hadoop Distributed File System, Distributed Hash Table, ASIC resistance, Turing Complete. Cryptography: Hash function, Digital Signature - ECDSA, Memory Hard Algorithm, Zero Knowledge Proof.

Unit 2

Blockchain: Introduction, Advantage over conventional distributed database, Blockchain Network, Mining Mechanism, Distributed Consensus, Merkle Patricia Tree, Gas Limit, Transactions and Fee, Anonymity, Reward, Chain Policy, Life of Blockchain application, Soft & Hard Fork, Private and Public blockchain.

Unit 3

Distributed Consensus: Nakamoto consensus, Proof of Work, Proof of Stake, Proof of Burn, Difficulty Level, Sybil Attack, Energy utilization and alternate. **Cryptocurrency**: History, Distributed Ledger, Bitcoin protocols - Mining strategy and rewards, Ethereum - Construction, DAO, Smart Contract, GHOST, Vulnerability, Attacks, Sidechain, Namecoin

Unit 4

Cryptocurrency Regulation: Stakeholders, Roots of Bitcoin, Legal Aspects - Cryptocurrency Exchange, Black Market and Global Economy

Unit 5

Blockchain Applications: Internet of Things, Medical Record Management System, Domain Name Service and future of Blockchain.

OUTCOMES:

At the completion of the course, the learner will be able to

- To know about cryptography and Digital Signatures.
- Gain more knowledge on the impact of Global Market due to cryptocurrencies.

References

1. Joseph Bonneau and Arvind Narayanan, *Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction*, First Edition, Princeton University Press, 2016.
2. Roger Wattenhofer, *The Science of the Blockchain*, First Edition, CreateSpace 2017
3. Andreas S. Antonopoulos, *Mastering Bitcoin: Unlocking Digital Cryptocurrencies*., First Edition, O'Reilly, 2014

4. Arvind Narayanan and Joseph Bonneau, *Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction*, First Edition, Princeton University Press, 2016

E3 CLOUD COMPUTING

Elective **Hours: 4** **Credits: 3**

OBJECTIVES:

- To understand the concept of cloud computing.
- To appreciate the evolution of cloud from the existing technologies like Virtualization.
- To have knowledge on the various issues in cloud computing.
- To be familiar with the lead players in cloud.
- To study the various security issues in cloud computing.

Unit 1

Introduction and Architecture: Technologies for Network-Based System: CPU and GPU – Software Environment for Cloud Computing: Service-Oriented Architecture (SOA) - Cloud Types : IaaS – PaaS - SaaS – Public - Private and Hybrid clouds – Characteristics of Cloud Computing - NIST Cloud Computing Reference Architecture.

Unit 2

Virtualization : Basics of Virtualization - Types of Virtualization - Implementation Levels of Virtualization - Virtualization Structures - Tools and Mechanisms - Virtualization of CPU – Memory - I/O Devices - Virtual Clusters and Resource management.

Unit 3

Cloud Infrastructure : Architectural Design of Compute and Storage Clouds - Layered Cloud Architecture Development – Design Challenges - Inter Cloud Resource Management – Resource Provisioning and Platform Deployment – Global Exchange of Cloud Resources.

Unit 4

Programming Model : Parallel and Distributed Programming Paradigms – MapReduce , Twister and Iterative MapReduce – Hadoop Library from Apache – Mapping Applications - Programming Support - Google App Engine, Amazon AWS - Cloud Software Environments - Eucalyptus, OpenStack, CloudSim

Unit 5

Cloud Security : Security Overview – Cloud Security Challenges – Software-as-a-Service Security – Security Governance – Risk Management – Security Monitoring – Security Architecture Design – Data Security – Application Security – Virtual Machine Security.

OUTCOMES*:*

On completion of the course, the learner will be able to

- Articulate the main concepts, key technologies, strengths and limitations of cloud computing.
- Learn the key and enabling technologies that help in the development of cloud.
- Develop the ability to understand and use the architecture of compute and storage cloud, service and delivery models.
- Explain the core issues of cloud computing such as resource management and security.

References

1. Kai Hwang, Geoffrey C Fox, Jack G Dongarra, *Distributed and Cloud Computing, From Parallel Processing to the Internet of Things*, Morgan Kaufmann Publishers, 2012.
2. Barrie Sosinsky, *Cloud Computing Bible*, Wiley India Pvt. Ltd, 2013.
3. John W.Rittinghouse and James F.Ransome, *Cloud Computing: Implementation, Management, and Security*, CRC Press, 2010.
4. Roger Jennings, *Cloud Computing with Windows Azure Platform*, Wiley India Pvt. Ltd, 2009.

E4 COMPUTER NETWORKS AND SECURITY

Elective **Hours: 4** **Credits: 3**

OBJECTIVES:

- To understand the network layers and its protocols.
- Implementation of Routing algorithms.
- To understand Networking through Socket Programming in Java.
- To know the importance of Network and Wireless Network security.
- To gain knowledge in Network threats and its preventive measures

Unit 1

Introduction: Physical and Data Link layer: Network H/W and S/W – Network models - Transmission media – Framing - **Network Layer:** Routing algorithm – Basic, Shortest path routing, Flooding, Distance vector routing, Link state routing, Routing for Mobile Host, IP address

Unit 2

Transport Layer: Transport service primitives, Elements of transport layer protocols-addressing, Connection establishment, Connection release, **Application Layer:** Electronic Mail(SMTP, MIME, POP3, IMAP), HTTP, WebServices, Name Service(DNS), FTP.

Unit 3

Network Protocols: UDP – Introduction, TCP protocol- Introduction, TCP header. **Socket Programming:** TCP, UDP Sockets, Client/server connection. Client Server Communication, File sharing. Implementation.

Unit 4

Network Security

Kerberos, X.509, PKI – Electronic Mail security – PGP – IP security – Web Security – SSL, TLS, SET.

Unit 5

Wireless Network Security

Wireless Network Security- IEEE 802.11 Wireless LANs - Protocol Overview and Security - Wireless Application Protocol (WAP) - Protocol Overview - Wireless Transport Layer Security (WTLS).

OUTCOMES:
At the completion of the course, the learner will be able to

- To design and understand the working of routing algorithms.
- Implementation of Socket Programming in UDP and TCP environment.
- Wide knowledge gained in wireless and network security.

References

1. Tanenbaum A. S. and David Wetherall, *Computer Networks*, Fifth Edition, Pearson, 2012.
2. Larry Peterson and Bruce Davie, *Computer Networks: A Systems Approach*, Fifth Edition, Elsevier Science, 2011
3. William Stallings, *Data and Computer Communications*, Eighth Edition, Prentice-Hall,2000.
4. James F. Kurose and Keith W. Ross, *Computer Networking: A Top-Down Approach Featuring the Internet*, Third Edition, Addison Wesley, 2005
5. Behrouz A. Forouzan, *Data Communications and Networking*, Second Edition, Tata McGraw-Hill, 2003.
6. Charles P. Pfleeger, Shari Lawrence Pfleeger, *Security in computing*, Third Edition, Prentice Hall, 2006.

E5 CONTENT MANAGEMENT SYSTEMS

Elective **Hours: 4** **Credits: 3**

OBJECTIVES:

- To teach and train the students the concepts of Web development
- To train the students to apply the concepts learnt through a mini project

Unit 1

Introduction: **Content Management Systems** - Overview of CMS based web sites and technologies - Developing Dynamic Web Content Sites - Setting site goals; Identifying target audiences - Wireframing - Planning site function and flow - Installing CMS applications - ISPs to add site features to servers - Working with MySQL and backend data structures.

Unit 2

Building and Administrating a website using **Wordpress** - Setting up and installing a Wordpress site - Finding and adding templates to a new site - Customize site features - Overview of administrative functions - Wordpress blogs - Case study

Unit 3

Web Site Design Using CSS - Overview of CSS features - Using CSS to redesign text features- Using CSS to move and position web graphics - Redesigning a web site: Alter the site's CSS tags. – Case study

Unit 4

Drupal: Overview - Drupal Site Building - Introduction to Drush - Drupal Module Development Drupal Database API Basics - Drupal Security Guidelines - Drupal Performance - Drupal Theming - Drupal Debugging

Unit 5

Security Updates and Applying Patches - Content Architecture - Advanced Theming - Drupal SEO - Drupal Accessibility – Form API, Field API - File Handling in Drupal - Drupal Upgrade - Drupal Commerce – Case study on Drupal

OUTCOMES:

At the completion of the course, the learner will be able to

- Understand the purpose and basics of a Content Management system
- Design webpage using CSS
- Build dynamic website using Server side scripting language
- Understand the concept of Database
- Create websites with Word Press
- Create website using Drupal

References

1. Boiko, Bob Content Management Bible. Wiley. p. 1176. Rockley, *Ann Managing Enterprise Content: A Unified Content Strategy*, New Riders Press, 2004.
2. Hackos, JoAnn T. *Content Management for Dynamic Web Delivery*, Wiley Publications, 2002.
3. Lisa Sabin, *WordPress For Dummies* , Seventh Edition, Wiley.
4. Daniel Sipos, *Drupal 8 module development*, Second Edition,, Packt Publishing in 2019-03 (580 pages), ISBN: 9781789612363
5. Preston So, *Decoupled Drupal in Practice*, Apress, 2018-12, ISBN: 9781484240717
6. J. Ayen Green, *Drupal 8 Quick Start Guide*, *Get up and running with Drupal 8*

E6 CRYPTOGRAPHY AND NETWORK SECURITY

Elective **Hours: 4** **Credits: 3**

OBJECTIVES:

- To understand the basics of Cryptography.
- To learn Encryption and Decryption algorithms.
- To learn the awareness of Digital attacks.
- To understand the importance of Network and System security.

Unit 1

Fundamentals And Mathematics Of Cryptography Overview - Classical Crypto Systems – Substitution Ciphers –Transposition Ciphers - Stream and Block Ciphers – Introduction to Number Theory – Congruences –Modular Arithmetic - Modular Exponentiation – Fermats and Eulers Theorem .

Unit 2

Encryption Techniques: Data Encryption Standard – Advanced Encryption Standard – Confidentiality using Symmetric Encryption - Public-Key Cryptography and RSA – Key Management - Diffie-Hellman Key Exchange – Elliptic Curve Cryptography – Symmetric Key Distribution – Kerberos - X.509 Authentication Service.

Unit 3

Hash Functions And Signatures: Message Authentication and Hash Functions – Description of MD Hash Family – Secure Hash Algorithms – SHA-512.

Unit 4

Digital Signatures and Authentication Protocols: – Digital Signature Standard – Process - Services - Attacks on Digital Signature - Digital Signature Schemes.

Unit 5

Network Security: Security at the application layer - E-Mail - Pretty Good Privacy – S/MIME – Security at the transport layer- Security at the Network Layer - IPSec – Security Policy – Security Association

System security:– Intrusion Detection and prevention –Malwares and Related Threats – DOS Attacks Firewalls – Firewall Types-Configuration and Implementation - Firewall Forensics - Services and Limitations

OUTCOMES:

At the completion of the course, the learner will be able to

- Knowledge gained on Encryption and Decryption algorithms.
- Gain more knowledge in Digital signature and hashing.
- Awareness of network threats and its possible security measures are understood.

References

1. William Stallings, *Cryptography And Network Security – Principles and Practices,* Sixth Edition, Pearson Education, 2013
2. Keith Martin, *Everyday Cryptography: Fundamental Principles and Applications,* Second Edition, Oxford, 2017
3. Behrouz A. Forouzan and Debdeep Mukhopadhyay, *Cryptography and Network Security*, Second Edition, Tata Mc Graw Hill, 2010.
4. Man Young Rhee, *Internet Security: Cryptographic Principles", "Algorithms and Protocols*, Third Edition, Wiley Publications, 2003.
5. Joseph Migga Kizza, *Guide to Computer Network Security*, Second Edition, Springer, 2010.
6. Charlie Kaufman, Radia Perlman and Mike Speciner, *Network Security,* Second Edition, Addison Wesley, 2002.

E7 CYBER SECURITY AND FORENSICS

Elective **Hours: 4** **Credits: 3**

OBJECTIVES:

- To bring awareness to students on the numerous possible computer-related crimes
- To present the existing laws and methods of countering the crimes
- To ignite the creative thinking of students to analyze an existing Cybercrime case and present a solution alternative, if any through a Case study

Unit 1

Cyber Crime - Internet: Introduction, Categories and Classifications of Cybercrimes: E-Mail Spoofing, Spamming, Cyber defamation, Internet Time Theft, Newsgroup Spam/Crimes from Usenet Newsgroup, Industrial Spying/Industrial Espionage, Hacking, Online Frauds,

Pornographic Offenses, Software Piracy, Password Sniffing, Credit Card Frauds and Identity Theft. **Cyber offenses**: Passive Attack, Active Attacks, Scanning/Scrutinizing gathered Information, Attack (Gaining and Maintaining the System Access), Social Engineering, Cyberstalking, Cybercafe and Cybercrimes, Botnets: The Fuel for Cybercrime, Attack Vector and Cloud Computing.

Unit 2

Cybercrime - Mobile and Wireless Devices: Introduction, Proliferation of Mobile and Wireless Devices, Trends in Mobility, Credit Card Frauds in Mobile and Wireless Computing Era, Security Challenges Posed by Mobile Devices, Registry Settings for Mobile Devices, Authentication Service Security, Attacks on Mobile/Cell Phones, Mobile Devices: Security Implications for Organizations, Organizational Measures for Handling Mobile, Organizational Security Policies and Measures for Mobiles and Laptops

Unit 3

Tools and Methods Used in Cybercrime: Introduction, Proxy Servers and Anonymizers, Phishing, Password Cracking, Keyloggers and Spywares, Virus and Worms, Trojan Horses and Backdoors, Steganography, DoS and DDoS Attacks, SQL Injection, Buffer Overflow, Attacks on Wireless Networks. Phishing, Identity Theft: Types of Identity Theft, Techniques of ID Theft, Identity Theft Countermeasures, Protection of Online Identity.

Unit 4

Cybercrimes and Cybersecurity: The Legal Perspectives: Introduction, Need for Cyberlaws: The Indian Context, The Indian IT Act, Challenges to Indian Law and Cybercrime Scenario in India, Consequences of Not Addressing the Weakness in Information Technology Act , Amendments to the Indian IT Act, Cybercrime and Punishment, Cyberlaw, Technology and Students: Indian Scenario.

Unit 5

Computer Forensics: Introduction, Digital Forensics Science: Need – Cyber forensics and Digital Evidence, Network forensics, Computer Forensics Investigation, Computer Forensics Laboratory Requirements, Computer Forensics and Steganography, Relevance of the OSI 7 Layer Model to Computer Forensics, Forensics and Social Networking Sites: The Security/Privacy Threats, Challenges, Special Tools and Techniques, **Forensics of Hand-Held Devices**: Hand-Held Devices and Digital Forensics, **Toolkits for Hand-Held Device Forensics**: EnCase, Device Seizure and PDA Seizure, Palm DD, Forensics Card Reader, Cell Seizure, MOBILedit!, ForensicSIM, Organizational Guidelines on Cell Phone Forensics.

OUTCOMES:

At the completion of the course the learner will be able to

- Increase awareness on Cyber crimes that affect personal and social life of a person
- Provide knowledge on ways of preventing oneself through Cyber Security measures
- Understand the field of Cyber Forensics and its benefits
- Equip the student to take up a career in cyber forensics and / or Cyber security
- Apply the concepts of cyber security and forensics through a project and provide innovative solutions

References

1. Nina Godbole, Sunit Belapur, *Cyber Security Understanding Cyber Crimes, Computer Forensics and Legal Perspectives*, Wiley India Publications, 2011
2. James Graham, Richar Howard,Ryan Olson, *Cyber Security Essentials*, CRC Press, Tailor and Francis Group, 2011
3. Robert Jones, *Internet Forensics: Using Digital Evidence to Solve Computer Crime*, O'Reilly Media, 2005
4. Chad Steel, *Windows Forensics: The field guide for conducting corporate computer investigations*, Wiley India Publications, 2006

E8 DATA WAREHOUSING AND MINING

Elective **Hours: 4** **Credits: 3**

OBJECTIVES:

- To train the students to understand data mining principles and techniques.
- To expose the students to the concepts of data warehousing
- To study algorithms for finding the hidden interesting patterns in data.
- To learn various Data Mining techniques such as classification, clustering and Association rule mining.

Unit 1

Data Mining & Data Preprocessing: Introduction to KDD process – Knowledge Discovery from Databases - Need for Data Preprocessing – Data Cleaning – Data Integration and Transformation – Data Reduction – Data Discretization and Concept Hierarchy Generation.

Unit 2

Data Warehouse : Data Warehousing - Operational Database Systems vs Data Warehouses - Multidimensional Data Model - Schemas for Multidimensional Databases – OLAP operations – Data Warehouse Architecture – Indexing – OLAP queries & Tools.

Unit 3

Association Rule Mining: Introduction - Data Mining Functionalities - Association Rule Mining - Mining Frequent Itemsets with and without Candidate Generation - Mining Various Kinds of Association Rules..

Unit 4

Classification & Prediction: Classification vs Prediction – Data preparation for Classification and Prediction – Classification by Decision Tree Introduction – Bayesian Classification – Rule Based Classification. Prediction – Accuracy and Error Measures – Evaluating the Accuracy of a Classifier or Predictor.

Unit 5

Clustering: Cluster Analysis - Types of Data in Cluster Analysis – Clustering Methods: Partitioning Methods – Hierarchical methods – Density-Based Methods. Clustering High-Dimensional Data. Outlier Analysis.

OUTCOMES:

At the completion of the course, the learner will be able to

- Preprocess the data to generate quality data for data mining applications.

- Evolve Multidimensional Intelligent model from typical system.
- Apply the association rules for mining the data.
- Design and deploy classification and clustering techniques.
- Evaluate various mining techniques on complex data objects.

References

1. Jiawei Han and Micheline Kamber, *Data Mining Concepts and Techniques*, Second Edition, Elsevier, Reprinted 2011.
2. K.P. Soman, Shyam Diwakar and V. Ajay, *Insight into Data mining Theory and Practice*, Easter Economy Edition, Prentice Hall of India, 2006.
3. G. K. Gupta, *Introduction to Data Mining with Case Studies*, Easter Economy Edition, Prentice Hall of India, 2006.
4. Pang-Ning Tan, Michael Steinbach and Vipin Kumar, *Introduction to Data Mining*, Pearson Education, 2007.

E9 DIGITAL IMAGE PROCESSING

Elective **Hours: 4** **Credits: 3**

OBJECTIVES:

- To provide the basic knowledge on image processing techniques like image acquisition, enhancement, transform. Restoration, segmentation, compression and object recognition in images and their applications.
- To impart the mathematical logic behind the various image processing algorithms
- To facilitate the students apprehend and implement various image processing algorithms.

Unit 1 - Digital Image Fundamentals, Image Acquisition and Image Enhancement

Digital Image Fundamentals: Digital Images and their Structure, Fields that use Digital Image Processing, Steps in Digital Image Processing, Components of an Image Processing System. *Image Acquisition:* Image Sensors and Scanners, Image Digitization - Sampling and Quantization, Image Types and Formats, Color Images, Color Models. *Image Enhancement:* Spatial Filtering – histogram processing and equalization, color image histogram, point, mask and neighborhood operations, Frequency Domain Filtering – low-pass and high-pass filters, Gamma Correction, Linear and Non-linear Grey Level Transformation, Image Arithmetic

Unit 2 - Image Transforms and Morphological Image Processing

Image Transforms: Need for Transforms, Fourier Transform, Discrete Fourier Transform, Walsh, Haar, Discrete Cosine Transform, Wavelet Transforms, Basic Geometric Transformations. *Morphological Image Processing:* Morphological Operations – dilation, erosion, open and close, hole-filling, convex hull, thinning, thickening, skeleton.

Unit 3 – Image Restoration and Compression

Image Restoration: Image Degradation Model, Noise Models, Spatial Filtering, Periodic Noise and Frequency Domain Filtering, Inverse Filtering, Weiner Filtering.

Image Compression: Redundancy in Images, Run-length Coding, Huffman Coding – binary and non-binary, Lossless and Lossy Compression, Image Compression Standards.

Unit 4 - Image Segmentation
Thresholding – global thresholding, adaptive thresholding, threshold selection, Point, Line and Edge Detection, Edge Linking and Boundary Detection – polygonal fitting, Hough Transform, Region Based Segmentation, Segmentation using Morphological Watersheds, Active Contour – greedy-snake algorithm,

Unit 5 - Object Recognition in Images
Patterns and Pattern Classes, Pattern Matching – minimum distance classifier, matching by correlation, Feature Extraction and Machine Learning for Object Recognition, Recognition Strategies - classification (Nearest Neighbor, Bayes, Neural Networks), Template Matching, Brief Introduction to Object Recognition using Deep Learning and Convolution Neural Networks.
Case Study – Performing image preprocessing using enhancement methods, morphology based segmentation to detect objects in the images and recognizing them using Python, ML/DL libraries and predefined models.

OUTCOMES:
- Ascertain and describe the basics of image processing concepts through mathematical interpretation.
- Analyze, design and implement the image processing algorithms studied for various real-time applications.
- Familiar with the use of Python and OpenCV for Image Analysis.

References

1. Rafael C Gonzalez, Richard E Woods, *Digital Image Processing*, Fourth Edition, Pearson Education, 2018
2. S. Jayaraman, S. Esakkirajan, T. Veerakumar, *Digital Image Processing*, Mc-Graw Hill, 2012
3. A.K. Jain, *Fundamentals of Digital Image Processing*, PHI, 2011
4. Mark Nixon, Alberto Aguado, *Feature Extraction and Image Processing*, Second Edition, Elsevier, 2008
5. Scott E Umbaugh, *Computer-Imaging: Digital Image Analysis and Processing*, CRC Press, 2000
6. Himanshu Singh, *Practical Machine Learning and Image Processing: For Facial Recognition, Object Detection, and Pattern Recognition Using Python*, Apress, 2019
7. Xiaoyue Jiang, Abdenour Hadid, Yanwei Pang, Eric Granger, Xiaoyi Feng (Eds.), *Deep Learning in Object Detection and Recognition*, Springer, Singapore, 2019
8. https://docs.opencv.org/master/d2/d96/tutorial_py_table_of_contents_imgproc.html
9. https://www.pyimagesearch.com/

E10 DIGITAL MARKETING

Elective **Hours: 4** **Credits: 3**

OBJECTIVES:
- To learn how to take a systematic approach to develop a Digital Marketing strategy
- To know the various Digital Analytic tools with overall marketing objectives

- To learn global strategy in digital marketing.
- To learn all the essentials of mobile analysis.

Unit 1

Digital Marketing and Digital Media : Introduction to marketing-digital marketing and its principles- Digital media types – Owned and earned social metrics – Paid searches and Organic Searches - Aligning Digital and Traditional Analytics – Identifying social media listening tools – Understanding social media engagement software – Social media engagement tools.

Unit 2

Tools For Digital Analytics : Social Media Listening Tools - Evolution, Social analytics life cycle, Social media monitoring software: Sysomos, Radian6, Visible Technologies, Zoho social and others. Search Analytics Tools – Basics of search, Search analytics use cases, Search data, Google trends, YouTube trends, Google Adwords keyword, Yahoo clues, Collecting insights through search data. Audience Analysis Tools – Audience Analysis Use Cases, Audience analysis tool types – Audience analysis Techniques, Event Triggers. Content Analysis Tools - Content Audits-Optimizing Content Distribution, Analysing Content Consumption. Engagement Analysis Tools

Unit 3

Digital Influence And Listening : Online advertising-Reality of Digital Influence - Media List - Klout, PeerIndex - Online Versus Offline Influence - Using the Influencer List - Developing Social Media Listening Program - Using Listening Data for Program Planning - Conversation Audit - Online Influencers - Conducting Social brand benchmarking - Use of Online data for crisis anticipation - Identifying known issues - Crisis day monitoring and ongoing reporting - Corrections after crisis - Improving customer service - Social customer service conflict - Social customer service models.

Unit 4

Research Plan And Search Analysis : Launching new product – Product life cycle – Introduction Phase – Growth Phase – Maturity Phase. Formulating research plan – Developing source list – Research methods – Constructing reports – Delivering reports – Report use cases – Building central repository of information – Search analytics for digital strategy – Search analytics for content strategy and planning – Optimizing content for search engines.

Unit 5

ROI and Mobile Analytics: Return on Investment (ROI) – Return on Engagement, Influence, Experience – Tracking ROI – Understanding measurement fundamentals – Measurement reporting cadence - Mobile Analytics – Mobile market landscape – Mobile marketing measurement – Marketing activities – Audience/visitor metric – Mobile app performance - Social CRM – Social CRM initiative – Social CRM Initiative - Future of Digital Data

OUTCOMES*:*

At the completion of the course, the learner will be able to

- Relate to digital media marketing and the need for analytics on the data captured.
- Choose the appropriate tools for performing different digital analytics on the digital marketing data.
- Analyze and appraise the outcomes of digital influence and listening.
- Formulate a research plan and perform search analysis on the digital marketing data.
- Summarize the strategies for Mobile analytics and Business Intelligence

References

1. Chuck Hemann and Ken Burbary, *Digital Marketing Analytics: Making Sense of Consumer Data in a Digital World*, First Edition, Que Publishing, ISBN-13: 978-0789750303, 2013.
2. Wayne L.Winston, *Marketing Analytics: Data driven techniques and Microsoft Excel*.
3. Simon Kingsnorth, *Digital Marketing Strategy: An Integrated Approach to Online Marketing*, First edition, Kogan Page Publisher, ISBN-13: 978-0749474706, 2016.
4. Dave Chaffey, Fiona Ellis-Chadwick, *Digital Marketing – Strategy, Implementation and Practice*, Sixth edition, Pearson Education, ISBN-13: 978-1292077611, 2016.
5. Calvin Jones, *The best digital marketing campaigns in the world, Mastering The Art of Customer Engagement*
6. Eric Enge, Andy Crestodina, Larry Kim, Steve Rayson and Chad White, *How the Pros Turn Marketing Analytics Into Effective Marketing Strategies*, Alexa, An Amazon Company.
7. blog.alexa.com/wp-content/uploads/2016/12/How-to-Pros-Turn-Marketing-Analytics-into-Effective-Marketing-Strategies-ebook.pdf

E11 ETHICAL HACKING

Elective **Hours: 4** **Credits: 3**

OBJECTIVES:

- To learn various hacking techniques and attacks.
- To know how to protect Windows and Networks
- To know how to protect data assets against attacks from the Internet.
- To assess and measure threats to information assets.
- To evaluate where information networks are most vulnerable.

Unit 1 - Introduction to Ethical Hacking and Methodology

Introduction to Ethical Hacking: Planning and performing attacks - Five stages of Ethical Hacking – Ways to conduct Ethical Hacking - Types of hacking – Hacker types

Hacking Methodology: Setting the stage for testing - Seeing what others see - Scanning systems - Determining what's running on open ports - Assessing vulnerabilities - Penetrating the system

Unit 2 - Reconnaissance and Social Engineering

Reconnaissance: HTTrack - Google Directives – Harvester - Whois – Netcraft –Host - Fierce and other tools to extract information from DNS - Extracting Information From E-mail Servers – MetaGooFil – ThreatAgent

Social Engineering: Implications of Social Engineering - Performing Social Engineering attacks – Social Engineering countermeasures

Unit 3 - Windows Hacking and Security

Security architecture of Windows – Windows User Account Architecture and Attack – Windows vulnerabilities – Detecting null sessions - Cracking BIOS and Windows Password – Changing Windows Visuals – Editing the OS – Registry - Other system files – Checking share permissions

Unit 4 – Network Hacking

Network infrastructure vulnerabilities - Scanning, poking and prodding the network - IP address – DNS – NslookUp – Port scanning and surfing - Sockets – PING – Netstat – Getting information about a domain – FTP Port and using FTP Client – FTP Commands –Detecting common Router, Switch and Firewall weaknesses – Discovering wireless network attacks and taking countermeasures -

Unit 5 – Web Hacking and Cracking Passwords

Web Hacking: Get, Post and Head methods – Hacking from Web Browser – Post Dial Up Screen hacking – Web security testing tools – Web Vulnerabilities – Minimizing Web security risks

Cracking Passwords: Password vulnerabilities – Cracking passwords – Password cracking countermeasures

OUTCOMES:

At the completion of the course, the learner will be able to

- Defend hacking attacks and protect data assets.
- Defend a computer against a variety of different types of security attacks using a number of hands-on techniques. .
- Practice and use safe techniques on the World Wide Web.

References

1. Ankit Fadia, *Ethical Hacking*, Second Edition, Macmillan India Ltd, 2006
2. Kevin Beaver, *Hacking for Dummies*, Fifth Edition, Wiley, 2016
3. *CEH official Certfied Ethical Hacking Review Guide*, Wiley India Edition, 2015.
4. Kenneth C.Brancik, I*nsider Computer Fraud*, Auerbach Publications Taylor & Francis Group, 2008.
5. Patrick Engebretson, *The Basics of Hacking and Penetration Testing*, Second Edition, Syngress, 2013

E12 FREE AND OPEN SOURCE SOFTWARE

Elective **Hours: 4** **Credits: 3**

OBJECTIVES:

- To teach the concepts, tool and practices of Open source software development
- To train the students in programming using Python language, which is widely used in the industry
- To train the students to apply the concepts of web development by building a website using Drupal Modules and Themes through a Case study

Unit 1

Introduction: FOSS community - Benefits of Community based Software Development - Need for free and Open source software – FOSS Programming guidelines and Practices - **Python**: Introduction - Basic Syntax, Variable and Data Types, Operator - Conditional Statements - Looping - Control Statements - String Manipulation - Lists - Tuple - Dictionaries - Functions - Modules - Input-Output - Exception Handling.

Unit 2

Advanced Python: Database: Introduction to MySQL, PYMYSQL Connections, Executing queries, Transactions, Handling error- Regular Expressions: Match function, Search function, Grouping, Matching at Beginning or End, Match Objects, Flags – Advanced concepts: Decorators, Generators, Iterators, Co-routines – CGI: Architecture, CGI environment variable, GET and POST methods, Cookies, File upload – Case study

Unit 3

Content Management System – Overview of CMS and CMS based websites and technologies - Git Basics – Programming: Best Practices – **Drupal**: Overview - Drupal Site Building

Unit 4

Introduction to **Drush** - Drupal Module Development Drupal Database API Basics - Drupal Security Guidelines - Drupal Performance - Drupal Theming - Drupal Debugging

Unit 5

Security Updates and Applying Patches - Content Architecture - Advanced Theming - Drupal SEO - Drupal Accessibility – Form API, Field API - File Handling in Drupal - Drupal Upgrade - Drupal Commerce – Case study

OUTCOMES:

At the completion of the course, the learner will be able to

- Acquire programming skills in core Python.
- Acquire Object Oriented Skills in Python
- Develop the skill of designing Graphical user Interfaces in Python
- Develop the ability to write database applications in Python
- Build Drupal based web applications using contributed modules.
- Develop website in Drupal using modules and themes

References

1. Kenneth A. Lambert, *Fundamentals of Python: First Programs*, CENGAGE Learning, 2012.
2. Paul Gries, Jennifer Campbell and Jason Montojo, *Practical Programming: An Introduction to Computer Science using Python 3*, Second edition, Pragmatic Programmers, 2013.
3. Timothy A. Budd, *Exploring Python*, Mc-Graw Hill Education (India) Private Ltd.,, 2015.
4. Jesús M. González-Barahona, Joaquín Seoane Pascual, Gregorio Robles, *Introduction to Free Software*, Free Technology Academy, 2009.
5. Angela Byron, Addison Berry, Bruno De Bondt, *Using Drupal*, Second Edition, O' Reilly, 2012
6. docs.python.org/2/tutorial/
7. sixrevisions.com/resources/git-tutorials-beginners/
8. groups.drupal.org/node/509541

E13 HEALTHCARE INFORMATION MANAGEMENT SYSTEMS

Elective **Hours: 4** **Credits: 3**

OBJECTIVES:

- To enable the learners to know the importance of Healthcare Information systems
- To teach the technology and method of developing Healthcare Information systems
- To teach the processing methods of Electronic Health Records
- To train the students in designing and developing Healthcare systems for a disease domain using case study

Unit 1

Introduction - Health IT - Introduction to Health Care Information Systems (HCIS) - Health care data quality – Health care information regulations, laws and standards.

Unit 2

Health Care Information Systems - History and evolution of health care information systems – Current and emerging use of clinical information systems – System acquisition – System implementation and support.

Unit 3

Information Technology - Information architecture and technologies that support health care information systems – Health care information system standards – Electronic health record (EHR) - Clinical point of care technology - Security of health care information systems – E-health

Unit 4

Management of IT Challenges - Organizing information technology services – IT alignment and strategic planning – IT governance and management.

Unit 5

IT Initiatives - Management's role in major IT initiatives – Assessing and achieving value in health care information systems. Case study - HCIS Development for a disease domain – Case study (Design and Development

OUTCOMES:

At the completion of the course, the learner will be able to

- Understand the importance of Healthcare system
- Design and develop a Healthcare information management system

References

1. Karen A Wager, Frances Wickham Lee, John P Glaser, *Managing Health Care Information Systems: A Practical Approach for Healthcare Executive*s, Second Edition, JohnWiley, 2009.
2. Marion J. Ball, Charlotte Weaver, Joan Kiel , *Healthcare Information Management Systems: Cases, Strategies, and Solutions*, Third Edition, Springer, 2010.
3. Rudi Van De Velde and Patrice Degoulet, *Clinical Information Sytems: A Componenet based approach*, Springer, 2005.
4. Kevin Beaver, *Healthcare Information Systems*, Second Edition, CRC Press, 2002
5. Marion J. Ball, *Healthcare Information Management Systems: A Practical Guid*e, Springer-Verlag GmbH, 1995

E14 INFORMATION SECURITY

Elective **Hours: 4** **Credits: 3**

OBJECTIVES:

- To understand Encryption and Decryption techniques in cryptography.
- To understand the importance of Security in Networks and Databases.
- To strengthen knowledge on Security Standards and Models.

Unit 1 : Elementary Cryptography

Terminology and Background – Substitution Ciphers – Transpositions Technique – Making Good Encryption Algorithms- Data Encryption Standard- AES Encryption Algorithm – Public Key Encryption – Key Exchange – Digital Signatures – Certificates

Unit 2 : Program Security

Secure programs – Non-malicious Program Errors – Malicious code – Countermeasures for Users and Developers – Control of Access to General Objects – User Authentication: Tokens and Biometrics – Good Coding Practices – Security in Operating Systems: Layered Design and Kernelized Design

Unit 3 : Security In Networks

Threats in networks – Encryption – Virtual Private Networks – PKI – SSH – SSL – IPSec – Content Integrity – Access Controls – Wireless Security – Honeypots – Traffic Flow Security – Firewalls – Intrusion Detection Systems – E-mail attacks.

Unit 4 : Security In Databases

Security requirements of database systems – Reliability and Integrity in databases – Redundancy – Recovery – Concurrency/ Consistency – Monitors – Sensitive Data – Types of disclosures –Inference-Finding and Confirming SQL injection

Unit 5 : Security Models And Standards

Secure SDLC – Secure Application Testing – Security architecture models – Trusted Computing Base – Bell-LaPadula Confidentiality Model – Biba Integrity Model – Graham-Denning Access Control Model – Harrison-Ruzzo-Ulman Model – Security Standards - ISO 27000 family of standards – NIST.

OUTCOMES:

At the completion of the course, the learner will be able to

- To know the importance of security in the digital world.
- To gain knowledge on Cryptography and its need.
- To know about Firewalls and various malicious Software.

References

1. Charles P. Pfleeger and Shari Lawrence Pfleeger, *Analyzing Computer security Computing-A threat*, Fourth Edition, Prentice hall, 2012.
2. Michael Whitman and Herbert J. Mattord, *Management of Information Security*, Third Edition, Cengage Learning, 2010.
3. William Stallings, *Cryptography and Network Security : Principles and Practices*, Fifth Edition, Prentice Hall, 2012.

4. Michael Howard, David LeBlanc and John Viega, *24 Deadly Sins of Software Security: Programming Flaws and How to Fix Them*, First Edition, Mc GrawHill Osborne Media, 2009.
5. Matt Bishop, *Computer Security: Art and Science*, First Edition, Addison-Wesley, 2002.

E15 INTERNET OF THINGS

Elective **Hours: 4** **Credits: 3**

OBJECTIVES:

- To know how IoT becomes the trend of future networking.
- To understand the need of IoT, WoT and CoT.
- To understand the various Protocols and standardizations used for IoT.
- To know about the role of IoT in user defined applications.
- To understand the Pros and Cons of IoT applications.

Unit 1

Introduction: Definitions and Functional Requirements –Motivation – Architecture – Four Pillars of IoT – DNA of IoT - The Toolkit Approach for End-user Participation in the Internet of Things. Middleware for IoT: Overview – Communication middleware for IoT

Unit 2

IOT Protocols: Protocol Standardization for IoT – Efforts – M2M and WSN Protocols – SCADA and RFID Protocols – Issues with IoT Standardization – Protocols – BACNet Protocol – Modbus – KNX – Zigbee Architecture

Unit 3

Web of Things: Web of Things versus Internet of Things – Two Pillars of the Web – Architecture Standardization for WoT– Platform Middleware for WoT – WoT Portals and Business Intelligence

Unit 4

Cloud & Integration: Cloud of Things: Grid/SOA and Cloud Computing – Cloud Middleware – Cloud Standards – Cloud Providers and Systems – Mobile Cloud Computing – The Cloud of Things Architecture - Integrated Billing Solutions in the Internet of Things Business Models for the Internet of Things

Unit 5

Applications: The Role of the Internet of Things for Increased Autonomy and Agility in Collaborative Production Environments - Smart Grid – Electrical Vehicle Charging

OUTCOMES:

At the completion of the course, the learner will be able to

- Get familiarized with various protocols and algorithms used in IoT.
- To learn about the impact of IoT in smart devices.
- Gain more knowledge in IoT devices and Sensors.
- Knowledge gained on Arduino and Raspberry pi products.

References

1. Honbo Zhou, *The Internet of Things in the Cloud: A Middleware Perspective*, Fourth Edition, CRC Press, 2012
2. Mark Harrison and Florian Michahelles, *Architecting the Internet of Things,* Second Edition Springer, 2011
3. Olivier Hersent, Omar Elloumi and David Boswarthick, *The Internet of Things: Applications to the Smart Grid and Building Automation* , First Edition, Wiley, 2012
4. Olivier Hersent, David Boswarthick, Omar Elloumi, *The Internet of Things – Key applications and Protocols*, First Edition, Wiley, 2012
5. Adrian McEwen and Hakim Cassimally *Designing the Internet of Things*, First Edition, Wiley, 2013.

E16 INTERNET OF THINGS AND CLOUD

Elective **Hours: 4** **Credits: 3**

OBJECTIVES:

- To know how IoT becomes the trend of future networking.
- To understand the various Protocols and standardizations used for IoT.
- To understand the Pros and Cons of IoT applications.
- To appreciate the evolution of cloud from the existing technologies like Virtualization.
- To be familiar with the lead cloud players and various issues in cloud.

Unit 1

Introduction: Definitions and Functional Requirements –Motivation – Architecture – Four Pillars of IoT – DNA of IoT - The Toolkit Approach for End-user Participation in the Internet of Things. Middleware for IoT: Overview – Communication middleware for IoT

Unit 2

IOT Protocols: Protocol Standardization for IoT – Efforts – M2M and WSN Protocols – SCADA and RFID Protocols – Issues with IoT Standardization – Protocols – BACNet Protocol – Modbus – KNX – Zigbee Architecture

Unit 3

Web of Things: Web of Things versus Internet of Things – Two Pillars of the Web – Architecture Standardization for WoT– Platform Middleware for WoT – WoT Portals and Business Intelligence

Unit 4

Introduction and Virtualization: IOT and Cyber-Physical Systems - Technologies for Network-Based System: CPU and GPU – Types of Virtualization – Virtualization Structures: Tools and Mechanisms - Virtualization of CPU and Memory - Cloud Computing Service Models

Unit 5

Parallel Computing: – MapReduce – Hadoop Library – Cloud Platforms: Amazon, AWS

Applications: Cloud trends in Ubiquitous Computing – Technologies for IOT: RFID, Sensor Networks and GPS – Applications of IOT: Retailing and Supply-chain Management, Smart Buildings

OUTCOMES*:*
On completion of the course, the learner will be able to

- Get familiarize with various protocols and algorithms used in IoT.
- To learn about the impact of IoT in smart devices.
- Gain more knowledge in IoT devices and Sensors.
- Learn the key and enabling technologies that help in the development of cloud.
- Explain the core issues of parallel computing and also the recent trends in cloud computing.

References

1. Honbo Zhou, *The Internet of Things in the Cloud: A Middleware Perspective*, Fourth Edition, CRC Press, 2012
2. Mark Harrison and Florian Michahelles, *Architecting the Internet of Things,* Second Edition Springer, 2011
3. Olivier Hersent, Omar Elloumi and David Boswarthick, *The Internet of Things: Applications to the Smart Grid and Building Automation* , First Edition, Wiley, 2012
4. Adrian McEwen and Hakim Cassimally *Designing the Internet of Things*, First Edition, Wiley, 2013.
5. Kai Hwang, Geoffrey C Fox, Jack G Dongarra, Distributed and Cloud Computing, From Parallel Processing to the Internet of Things, Morgan Kaufmann Publishers, 2012.
6. Barrie Sosinsky, *Cloud Computing Bible*, Wiley India Pvt. Ltd, 2013.
7. Roger Jennings, *Cloud Computing with Windows Azure Platform*, Wiley India Pvt. Ltd, 2009.

E17 MACHINE LEARNING TECHNIQUES

Elective **Hours: 4** **Credits: 3**

OBJECTIVES:

- To understand the concepts and techniques of Machine Learning.
- To understand the Supervised and Unsupervised learning techniques
- To study the various probability based learning techniques
- To understand graphical models of machine learning algorithms

Unit 1

Introduction to Machine Learning: Learning – Types of Learning - Need for machine Learning - Fundamental concepts and applications of machine learning - Advantages and shortcomings of widely used machine learning algorithms Mathematical foundations of machine learning - random variables and probabilities - Probability Theory – Probability distributions - Decision Theory - Information Theory

Unit 2

Supervised Learning: **Decision Trees, Naïve Bayes Classification, Ordinary Least Squares Regression, Logistic Regression,** Probabilistic Generative Models - Probabilistic Discriminative Models - Support **Vector Machines** - Ensemble methods – Bagging - Boosting

- Neural Networks: Feed-forward Network Functions, Back- propagation - Deep learning networks

Unit 3

Unsupervised Learning: Clustering - K-means - EM Algorithm- Mixtures of Gaussians - The Curse of Dimensionality - Dimensionality Reduction - Factor analysis - Principal Component Analysis - Probabilistic PCA Independent component analysis

Unit 4

Advanced Learning: Undirected and directed graphical models - Bayesian networks and Markov Random Fields Sampling Reinforcement Learning - K-Armed Bandit Elements - Model-Based Learning- Value Iteration - Policy Iteration - Temporal Difference Learning Exploration Strategies - Deterministic and Non-deterministic Rewards and Actions - Computational Learning Theory - Mistake bound analysis, sample complexity analysis, VC dimension - Occam learning, accuracy and confidence boosting

Unit 5

Machine Learning Tools and Applications: Usage of tools – Machine learning platform: WEKA machine learning workbench, R platform, Python Scipy - Machine Learning Library: scikit-learn in Python, JSAT in Java, Accord Framework in .NET – GUIs: KNIME, RapidMiner, Orange – Applications: Prediction using data, Speech recognition, Healthcare, Object recognition in images, Natural Language Processing, Online search

OUTCOMES:

At the completion of the course, the learner will be able to

- Distinguish between, supervised, unsupervised and semi-supervised learning
- Apply the apt machine learning strategy for any given problem
- Suggest supervised, unsupervised or semi-supervised learning algorithms for any given problem
- Design systems that uses the appropriate graph models of machine learning
- Modify existing machine learning algorithms to improve classification efficiency

References

1. Christopher Bishop, *Pattern Recognition and Machine Learning*, Springer, 2007
2. Kevin P. Murphy, *Machine Learning: A Probabilistic Perspective*, MIT Press, 2012
3. Ethem Alpaydin, *Introduction to Machine Learning*, Third Edition, MIT Press, 2014
4. Tom Mitchell, *Machine Learning*, McGraw-Hill, 1997
5. Trevor Hastie, Robert Tibshirani, Jerome Friedman, *The Elements of Statistical Learning*, Second Edition, Springer, 2011
6. Stephen Marsland, *Machine Learning - An Algorithmic Perspective*, Second Edition, Chapman and Hall/CRC Press, 2014.
7. Andreas C. Müller, Sarah Guido, *Introduction to Machine Learning with Python: A Guide for Data Scientists*, O'Reilly, 2017
8. John Paul Mueller, Luca Massaron, *Machine Learning for Dummies*, John Wiley and Sons, 2016
9. Brett Lantz, *Machine Learning with R*, Second Edition, PACKT Publishing, 2015

E18 MOBILE APPLICATION DEVELOPMENT

Elective **Hours: 4** **Credits: 3**

OBJECTIVES :

- To understand the needs and characteristics of mobile applications.
- To understand mobile operating systems -Android and their architecture.
- To learn Android development tools.
- To work with Android User Interface Design and Event Handling.
- Master Android SQLite database and Content Provider.
- Gain knowledge to develop Gesture based UI, Multimedia, SMS, Phone Call, Sensor and Location Based Android application.

Unit 1

Introduction: Mobile Applications – Characteristics and Benefits – Application Model**, Android Introduction & Building Blocks:** Android Features**,** Architecture, Building Blocks, Application Components-**Development Environment:** Using Eclipse IDE for Mobile Development, Android Sdk, Emulator.

Unit 2

User Interface: User Interface Controls- **Activities**: Styles and Themes- Fragments-Intents: Linking Activities with Intent, Intent Filters-**View Groups**: Linear Layout, Table Layout, Relative Layout, Scroll View and Action Bar-**Views** : List View, Spinner View, Image View, Grid View- Event Handling- Gesture based UIs and UI using other platforms.

Unit 3

Dealing with Data: Database Connectivity, Content Providers-Data Persistence: Saving and loading user preferences, Persisting Data to files, Creating and using databases, Sharing Data using Content providers and Current trends.

Unit 4

Multimedia & Service Introduction: Multimedia: Working with Audio and Video- **Service:** Services and Background Processing, Binding Services, Communicating with services, Communication via the Web –Communication Methods (JSON)- **Application Development:** Graphics and Animation, Telephony, Notification and Alarms, Location based Application.

Unit 5

Android Services, Sensors: Introduction, Sensor Manager, Sensor classes-**Bluetooth**: Managing the Local Bluetooth Device Adapter, Bluetooth communications-**Wi-Fi**: Monitoring Wi-Fi Connectivity, Transferring Data

OUTCOMES :

At the completion of the course, the learner will be able to

- Learn Android application run time environment.
- Design the right user interface for an Android application.
- Learn Android SQLite database connection.
- Develop Android applications using Android UI toolkits and frameworks.

References

1. Wei-Meng Lee, *Beginning Android 4 Application Development*, Wiley, 2012.

2. Onur Cinar, *Android Apps with Eclipse*,,Apress, Springer, 2012.
3. Reto Meier, *Professional Android 4 Application Development*, Second Edition, Wiley India Edition, 2014.
4. John Horton, *Android Programming for Beginners*, Second Edition, Packt, 2018.
5. developer.android.com/training/basics/firstapp/index.html
6. www.tutorialspoint.com/android/index.htm
7. www.javatpoint.com/android-tutorial
8. www.vogella.com/articles/Android/article.html

E19 MOBILE COMMERCE TECHNOLOGY

Elective **Hours: 4** **Credits: 3**

OBJECTIVES:

- To understand M-commerce services.
- To understand M -commerce infrastructure, applications and Mobile Marketing.
- To know the availability of the latest technology and applications of M- commerce in various domains.
- Learn how to deal with business-to-business applications.
- Understand online business activities through mobile devices – such as smartphones or tablet computers - across a wireless internet connection.

Unit 1

Introduction: M– Commerce – Types of Mobile Commerce Services – Technologies of Wireless Business – Benefits and Limitations, Support, Mobile Marketing & Advertisement, Non– Internet Applications in M– Commerce – Wireless/Wired Commerce Comparisons**.**

Unit 2

Mobile Marketing : Businesses Vs Mobile Marketing - Classic Mistakes in Mobile Marketing - Laying Foundation for Successful Mobile Marketing Campaign - Understanding Technology behind Mobile Marketing – Android – iOS - Windows Phone - Strategic thinking about Mobile Marketing Campaign

Unit 3

Mobile Commerce Technology: A Framework for the Study of Mobile Commerce – NTT Docomo's I– Mode – Wireless Devices for Mobile Commerce – Towards a Classification Framework for Mobile Location Based Services – Wireless Personal and Local Area Networks –The Impact of Technology Advances on Strategy Formulation in Mobile Communications Networks

Unit 4

Mobile Applications : Location Based Marketing – LBS – NFC - Bluetooth and LBA - 2D Codes – Tablet - Other Mobile Applications - Business Firms Connecting To Customers Using Mobile – Case Study - Mobile Marketing For B2B Companies - Mobile E-Commerce to Drive Revenue.

Unit 5

Mobile Application Development: Mobile Payments - Present and Future Mobile Technology - Mobile Application Development.

OUTCOMES:
At the completion of the course, the learner will be able to

- Learn how to apply M – commerce principles to various business domains.
- Understand the theory and applications of M-commerce in the business domain.
- Get an exposure to current technological advancements in M-commerce.
- Able to build M – commerce business models.

References

1. Jeanne Hopkins, Jamie Turner, *Go Mobile: Location-Based Marketing, Apps, Mobile Optimized Ad Campaigns, 2D codes and other Mobile Strategies to Grow your Business*, First Edition, Wiley, 2012.
2. Paul Skeldon, *M- Commerce*, Second Edition, Crimson Publishing, 2012.
3. Karabi Bandyopadhyay, *Mobile Commerce*, First Edition, PHI Learning Private Limited, 2013.
4. Paul May, Tom Jell, *Mobile Commerce, Opportunities, Applications and Technologies of Wireless Business*, First Edition, Cambridge University Press, 2001.
6. Brian E. Mennecke, Troy J. Strader, *Mobile Commerce: Technology, Theory and Applications*, First Edition, Idea Group Inc., IRM press, 2003.
7. www.tutorialspoint.com/mobile_marketing/m_commerce.htm

E20 MOBILE COMPUTING

Elective **Hours: 4** **Credits: 3**

OBJECTIVES :

- To understand the basic concepts of mobile computing.
- To learn the basics of mobile telecommunication systems.
- Master with Network layer protocols and Ad-Hoc networks.
- To know the basis of Transport and application layer protocols.
- Gain knowledge about different mobile platforms and application development.

Unit 1

Introduction: Mobile Computing – Mobile Computing Vs wireless Networking – Mobile Computing Applications – Characteristics of Mobile computing-**Wireless Communication Fundamentals :** Introduction – Wireless transmission – Frequencies for radio transmission – Signals – Antennas – Signal Propagation – Multiplexing – Modulations – Spread spectrum – MAC – SDMA – FDMA – TDMA – CDMA – Cellular Wireless Networks.

Unit 2

Telecommunications System: Introduction to Cellular Systems-GSM- Services and Architecture-Protocols-Connection Establishment-Frequency Allocation-Routing-Handover-Security-Gprs.

Unit 3

Wireless Networks : Wireless LAN-IEEE 802.11 Standards-Architecture-Services-HIPERLAN-AdHoc Network-BlueTooth

Unit 4

Mobile Network Layer : Mobile IP – Dynamic Host Configuration Protocol - Routing – DSDV – DSR – Alternative Metrics- Wireless Application Protocol-WAP

Unit 5

Mobile Ad hoc Networks (MANETs): Overview, Properties of a MANET, spectrum of MANET applications, routing and various routing algorithms, security in MANETs.

OUTCOMES:

At the completion of the course, the learner will be able to

- Acquire knowledge about various types of Wireless Data Networks.
- Understand architecture, challenges of Wireless Communication.
- Master Wireless Protocols to shape the future Internet.
- Learn different types of Wireless Communication Networks and their functionalities.
- Able to develop simple Mobile Application.

References

1. Prasant Kumar Pattnaik, Rajib Mall, *Fundamentals of Mobile Computing*, Second Edition, PHI Learning Pvt.Ltd,, 2016.
2. Dharma Prakash Agarval, Qing-An Zeng, *Introduction to Wireless and MobileSyst*ems,Third Edition, Thomson Asia Pvt Ltd, 2011.
3. Uwe Hansmann, Lothar Merk, Martin S. Nicklons and Thomas Stober, *Principles of Mobile Computing*, First Edition, Springer, 2003.
4. William.C.Y.Lee, *Mobile Cellular Telecommunications-Analog and Digital System*s, Second Edition,Tata Mc Graw Hill Edition ,2006.
5. William Stallings, *Wireless Communications and Networks*, Second Edition, Pearson, 2005.
6. Reza, Bfar, *Mobile Computing Principles Designing and Developing Mobile Applications with UML and XML*, First Edition, Cambridge University Press, 2005
7. www.tutorialspoint.com/mobile_computing/index.htm

E21 NATURAL LANGUAGE PROCESSING

Elective **Hours: 4** **Credits: 3**

OBJECTIVES:

- To enable the students to understand the basics of language structure and how it is interpreted by the computer
- To teach students to program in Python and develop a real-life application by processing text

Unit 1

Introduction: History, Overview and challenges of NLP - NLP tasks: Syntax, Semantics, and Pragmatics – **Language Modeling:** Grammar- based Language Models, Statistical Language Model - Simple N-gram models. Estimating parameters and smoothing, Evaluating language models.

Unit 2

Syntactic Analysis: Regular Expressions, Finite-State Automata **- Part of Speech Tagging:** Lexical syntax. Hidden Markov Models **- Syntactic parsing:** Grammar formalisms and treebanks. Context-free grammars - Statistical parsing and probabilistic CFGs, Dependency parsing, Morphological Parsing, Constituency-Parsing

Unit 3

Semantic Analysis: Meaning Representation – Ambiguity - Lexical semantics and Word-Sense Disambiguation, Relationship extraction. **Semantic analysis techniques**: Text classification: Topic classification, Sentiment analysis, Intent classification - **Information Extraction**: Named Entity Recognition, Relation Extraction, Template Filling - **Semantic Parsing**. Discourse Processing, Coreference Resolution

Unit 4

Applications: Question Answering: IR-based Factoid Question Answering, Knowledge-based Question Answering **Summarization:** Automatic text summarization, Text summarization features, Extractive text summarization techniques, Abstractive text summarization methods: Structured based approaches and Semantic Based Approaches - **Machine Translation**: Basic issues in MT. Statistical translation, Phrase-based translation

Unit 5

NLP Using Python – NLTK: Case study - Accessing Text Corpora and Lexical Resources - Processing Raw Text - Categorizing and Tagging Words – Classification of text - Extracting Information from Text

OUTCOMES:

At the completion of the course, the learner will be able to

- Understand the basics of Human language and computer language models
- Gain knowledge on various parsing techniques.
- Understand how computers understand meanings and processes text
- Appreciate the application of NLP in real-life applications
- Implement NLP approaches in a real-time project using Python NLTK

References

1. Jurafsky, D. and J. H. Martin. *Speech and language processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*, Third Edition, Prentice-Hall, 2019.
2. Steven Bird, S., Klein, E., Loper, E, *Natural Language Processing with Python - Analyzing Text with the Natural Language Toolkit*, O'Reilly Media, 2010.
3. Tanveer Siddiqui, U.S. Tiwary, *Natural Language Processing and Information Retrieval*, Oxford University Press, 2008.

E22 OBJECT ORIENTED SYSTEM DESIGN

Elective **Hours: 4** **Credits: 3**

OBJECTIVES:

- To understand the entire Life cycle of Object Oriented and System Design.

- To understand the elements of Object Model.
- To know about the Unified Modeling Language (UML diagrams).
- To develop a software model using OOA and OOD processes.
- To understand the importance of Software Quality Assurance Testing

Unit 1

Introduction: The object model: Elements of the object model - abstraction, encapsulation, modularity, hierarchy, typing, concurrency, persistence. Object-oriented systems development life cycle, Object-oriented methodologies, Patterns, Frameworks, The Unified approach.

Unit 2

Unified Modeling Language (UML): UML static modeling - Class diagram, Use case diagram. UML dynamic modeling - Interaction diagram, Sequence diagram, Collaboration diagram, State chart diagram, Activity diagram and Implementation diagrams – Component diagram and deployment diagram.

Unit 3

Object-oriented Analysis: Identifying Use cases – Use case model. Object Analysis: Classification – Noun Phrase Approach, Common class patterns approach, Use-case driven approach, Classes Responsibilities and Collaborators.

Identifying object relationships, attributes and methods – Associations, Super-Sub class relationships, Part-of relationships-Aggregation, Class responsibility-Identifying attributes and methods – Defining Attributes from UML diagrams, Object responsibility – methods and messages.

Unit 4

Object-Oriented Design: Object-oriented design process and design axioms, Designing classes – The process, class visibility, refining attributes, designing methods and protocols, packages and managing classes. Designing objects with responsibilities – Creator – Information expert – Low Coupling – High Cohesion – Controller - Design Patterns – creational - factory method - structural – Bridge – Adapter - behavioral – Strategy – observer

Unit 5

Software quality assurance: Testing Strategies**,** Impact of Object orientation on testing, Object-oriented metrics, Test cases, Test plan, Continuous Testing. **Case study.**

OUTCOMES:

At the completion of the course, the learner will be able to

- How to Analyze and Design Software products based on Class and Uml diagrams.
- Gain more knowledge in Analysis, Design and Testing of Software Projects.

References

1. Ali Bahrami, *Object oriented Systems Development*, Seventh Edition, Irwin McGraw-Hill, 1999
2. Grady Booch et al., *Object-Oriented Analysis and Design with applications*, Third Edition, Pearson Education, 2008
3. Craig Larman, *Applying UML and Patterns - An Introduction to Object-Oriented Analysis and Design and Iterative Development*, Third Edition, Pearson Education, 2008

4. Joseph Schmuller, *Sams Teach yourself UML in 24 hours*, Techmedia, First Edition, 1999
5. Matin Fowler and Kendal Scott , *UML Distilled A brief guide to Standard Object Modeling*, Second Edition, Addison Wesley, 2002

E23 PRINCIPLES OF OPERATING SYSTEMS

Elective **Hours: 4** **Credits: 3**

OBJECTIVES:

- To teach the structure, storage and processes using which an Operating System of a computer works
- To train the students to apply the concepts through system programming tutorials

Unit 1
Introduction: Operating System Structure - Operating System Operations - Operating system services, System calls, System programs, Operating system - Design and implementation, System Boot.

Unit 2
Process Management: Process concept, Process Scheduling, Interprocess communication – **Threads - Process Synchronization:** Critical-Section problem, Peterson's Solution, Mutex Locks, Semaphores, Classic problems of Synchronization - Bounded Buffer, The Readers-Writers Problem, The Dining Philosopher's problem

Unit 3
CPU Scheduling: Scheduling Criteria, Scheduling Algorithms - **Deadlocks:** Characterization, Methods for handling Deadlocks: Deadlock Detection, Deadlock Prevention, Deadlock Avoidance, Recovery from Deadlock.

Unit 4
Memory Management: Main Memory - Segmentation, Paging - **Virtual Memory:** Demand paging, Copy-on-write, Page replacement, Allocation of Frames, Thrashing - **Storage Management:** Mass Storage Structure - **File System Interface:** Directory, Disk Structure, File Sharing, Protection.

Unit 5
File System Implementation: Directory Implementation, Allocation Methods, Free-Space management - **Protection:** Goals, Principles and Domains of protection**,** Access Matrix, Access Control, Revocation of Access Rights - **Security:** Program threats, System and Network threats, User authentication, Firewall.

OUTCOMES :
At the completion of the course, the learner will be able to
- Understand design of an operating system and the services provided by the OS.
- Explain how processes are synchronized and scheduled.

- Understand the different approaches to memory management
- Apply and appreciate the structure and organization of the file system.
- Demonstrate the use of system calls for managing processes, memory and files
- Gain knowledge on protection and security mechanisms.

References

1. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, *Operating System Concepts*, Ninth Edition, Wiley, 2013.
2. Deitel H.M., *Operating Systems*, Second Edition, Addison Wesley Publishing Company, 2000.
3. Milan Milenkovic, *Operating System Concepts and Design*, Eighth Reprint, Tata McGraw-Hill, 2000.
4. Tanenbaum A. S. and Albert S. Woodhull, *Operating Systems: Design and Implementation*, Second Edition, Prentice-Hall of India, 2000.

E24 SOFT COMPUTING

Elective **Hours: 4** **Credits: 3**

OBJECTIVES:

- To impart knowledge on Soft Computing methods and practices
- To train the students to apply the concepts and practices through case study

Unit 1

Introduction to Soft Computing: Concept of computing systems - "Soft" computing versus "Hard" computing - Characteristics of Soft computing - Some applications of Soft computing techniques

Unit 2

Fuzzy logic: Introduction to Fuzzy logic - Fuzzy sets and membership functions - Operations on Fuzzy sets - Fuzzy relations, rules, propositions, implications and inferences - Defuzzification techniques - Fuzzy logic controller design - Some applications of Fuzzy logic

Unit 3

Genetic Algorithms: Concept of "Genetics" and "Evolution" and its application to proablistic search techniques - Basic GA framework and different GA architectures - GA operators: Encoding, Crossover, Selection, Mutation, etc. - Solving single-objective optimization problems using GAs.

Unit 4

Multi-objective Optimization Problem Solving: Concept of multi-objective optimization problems (MOOPs) and issues of solving them - Multi-Objective Evolutionary Algorithm - (MOEA) - Non-Pareto approaches to solve MOOPs - Pareto-based approaches to solve MOOPs

Some applications with MOEAs

Unit 5

Artificial Neural Networks: Biological neurons and its working - Simulation of biological neurons to problem solving - Different ANNs architectures - Training techniques for ANNs - Supervised Learning Neural Networks – Unsupervised Learning Neural Networks - Applications of ANNs to solve some real life problems.

OUTCOMES:

At the completion of the course, the learner will be able to

- Understand the basic concepts of the various components of soft computing
- Gain knowledge of Fuzzy Logic
- Design and develop Fuzzy system
- Acquire the basic knowledge on Genetic algorithms
- Apply the concepts of Neuro - Fuzzy Modeling.
- Gain basic knowledge on Artificial Neural Networks

References

1. Jyh-Shing Roger Jang, Chuen-Tsai Sun, Eiji Mizutani, *Neuro-Fuzzy and Soft Computing*, Prentice-Hall of India Learning, 2009.
2. Simon Haykin, *Neural Networks and Learning Machines*,Third Edition, PHI Learning, 2011.
3. George J. Klir and Bo Yuan, *Fuzzy Sets and Fuzzy Logic-Theory and Applications*, Prentice Hall, 1995.
4. James A. Freeman and David M. Skapura, *Neural Networks Algorithms, Applications, and Programming Techniques*, Pearson Edn., 2003.
5. David E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Lear*ning, Addison Wesley, 2007.
6. Rajasekaran, and G. A. Vijayalakshmi Pai, *Neural Networks, Fuzzy Logic and Genetic Algorithms : Synthesis, and Applications*, Prentice Hall of India, 2007.
7. Ross Timothy J, *Fuzzy Logic with Engineering Applications*, Wiley India Pvt Ltd,, 2010

E25 SOFTWARE ENGINEERING, PROJECT MANAGEMENT AND TESTING

Elective **Hours: 4** **Credits: 3**

OBJECTIVES:

- To present the traditional and contemporary software development process models
- To train the students to analyse and develop software development solutions using different process models
- To present the Industry relevant methods and practices in Software project management to improve their industry-readiness
- To impart sound knowledge in testing of software

Unit 1

Software Engineering: Concepts – Development activities – Software lifecycle models - Classical waterfall - Iterative waterfall – Prototyping – Evolutionary - Spiral – Reusable - RAD, **Software Requirement Specification:** Requirement analysis and specification – Functional modelling – Data Flow Diagram

Unit 2

Design: Design concepts – Coupling – Cohesion – Functional independence – **Design patterns** – Singleton, Factory, Strategy, Observer, Builder, Adapter, State, **Model-view-controller** Architectural styles – Layered - Client- server - Tiered - Pipe and filter- User interface design

Unit 3

Agile Development Methods: Overview, Agile Manifesto, Refactoring Techniques, Limitations of Agile Process - **Agile Modeling:** Principles, Comparing XP and Agile Modeling - Lean Software Development***:*** Scrum - Dynamic Systems Development Methodology - **Extreme Programming - AI-Driven Software Engineering**: Traditional software development Vs Machine learning model development, Design, Testing, Automatic code generation; AI Software Development Tools

Unit 4

Software Project Management – Project planning – Effort Estimation – *Activity planning:* Sequencing and Scheduling activities - Network planning models - Precedence networks - Activity on arrow networks - Critical path -*Resource Allocation* - *Monitoring and Control* - *Software Quality* - SQA plan – Software Configuration Management

Unit 5

Software Metrics: Measuring Process and Product attributes, Metrics for Analysis Model - Metrics for Design Model - **Software Quality Assurance Standards - Software Testing Approaches:** Black box Testing: Random Testing, Equivalence Class Partitioning, Boundary Value Analysis - White box Testing: Control Flow Graphs, Paths: **Testing Methods**

OUTCOMES:

At the completion of the course, the learner will be able to

- Acquire knowledge in traditional software development models
- Extract and analyze software requirements specifications for different projects
- Develop skills in applying MVC model in software development
- Acquire knowledge in recent software development models
- Conceptualize software development in terms of AI systems
- Gain knowledge about the phases in software project management
- Identify and use the appropriate metrics during software development

References

1. Roger Pressman and Bruce Maxim, *Software Engineering: A Practitioner's Approach*, Ninth Edition, 2020
2. Robert C. Martin, *Agile Software Development: Principles*, Patters and Practices
3. Kenneth S. Rubin, *Essential Scrum: A Practical Guide to the Most Popular Agile Process*, First Edition , 2012.

E26 SYSTEM SOFTWARE

Elective **Hours: 4** **Credits: 3**

OBJECTIVES:

- To understand Machine Language and instructions.

- To gain knowledge in Assemblers, Linkers and Loaders and Compilers.
- To know about different compilers and various system tools.
- To know about the Machine independent and dependent features regarding Assemblers and Loaders

Unit 1

System software and machine architecture, Simplified Instructional Computer (SIC): SIC machine architecture, SIC/XE machine architecture, SIC programming examples.

Unit 2

Assemblers: Basic functions: simple SIC assembler, assembler algorithms and data structures; Machine dependent assembler features: instruction formats and addressing modes, program relocation; Machine-independent assembler features: literals, symbol defining statements, expressions, program blocks, control sections and program linking; Design options: one-pass assemblers, multi-pass assemblers

Unit 3

Loaders and Linkers: Basic functions: absolute loader, simple bootstrap loader; Machine-dependent features: relocation, program linking, algorithms and data structures for a linking loader; Machine-dependent features: automatic library search, loader options; Design options: linkage editors, dynamic linking, bootstrap loaders

Unit 4

Basic compiler functions: grammars, lexical analysis, syntactic analysis, code generation; Machine-dependent compiler features: intermediate form of program, machine-dependent code optimization- Machine-independent compiler features: structured variables, machine-independent code optimization, storage allocation, block structured languages; Compiler design options: division into passes, interpreters, p-code compilers, compiler-compilers. **Macroprocessors**-MASM

Unit 5

System Software Tools

Text editors - Overview of the Editing Process - User Interface Editor Structure. Interactive debugging systems - Debugging functions and capabilities – Relationship with other parts of the system – User-Interface Criteria. Implementation

OUTCOMES:

On completion of the course, the learner will be able to

- To know Assemblers, Loaders and Compilers in detail.
- Gain more knowledge in Programming Assembly Language.
- To understand Macroprocessors.

References

1. Leland L. Beck, *System Software - An Introduction to System Programming*, Third Edition, Pearson Education, 2007
2. D. M. Dhamdhere, *Systems Programming and Operating Systems,* Second Edition, Tata McGraw-Hill, 1999.
3. John R. Levine, Linkers & Loaders, Third Edition, Elsevier science, 2000.
4. Andrew Appel and Maia Ginsburg, *Modern Compiler Implementation in C*, Second Edition, Cambridge University Press, 2004.

E27 USER INTERFACE AND USER EXPERIENCE DESIGN

Elective **Hours: 4** **Credits: 3**

OBJECTIVES:

- To create an awareness on the importance of UI in computer applications,
- To train the students applying technology and creativity in the design of UI
- To emphasize the design of UI for a good User experience
- To apply the concepts of UI and UX in a case study

Unit 1

Introduction: Human–Computer Interface – Types of UI: Command line (cli), graphical user interface (GUI). Menu driven (mdi), Form based (fbi), Natural language (nli)

Unit 2

Human Computer Interaction: User Interface Design Process – Obstacles –Usability – Human Characteristics In Design – Human Interaction Speed – Human Consideration In Screen Design – Structures Of Menus – Functions Of Menus–Contents Of Menu– Formatting – Phrasing The Menu – Selecting Menu Choice– Navigating Menus– Graphical Menus.

Unit 3

User Interface: Overview - Interface design, Quality, PLOA (Principles of Least Astonishment), Types – **UI Design**: Processes, Requirements, UI research – **Graphical User Interface:** Structural Elements: Windows, Menus, Icons, Controls, Tabs – Interaction elements: Cursor, Pointer, Insertion Point – **Web User Interface** –Popularity –Characteristic & Principles.

Unit 4

User experience: UX Design, User research, Content strategy - Transition from Defining to **Designing** – **Design Principles**: Visual design, Interaction – Site Maps & Task flows – Wireframes and Annotations – Prototyping – Design testing with Users – Transition from **Design to Development**

Unit 5

Lean UX: Introduction, Principles: The Three Foundations of Lean UX, Process: Vision, Framing, and Outcomes

OUTCOMES:

At the completion of the course, the learner will be able to

- Develop applications with a profession UI
- Develop applications with UI that provide a good UX

References

1. Wilbent. O. Galitz ,*The Essential Guide To User Interface Design*, John Wiley&Sons, 2001.
2. Ben Sheiderman, *Design The User Interface*, Pearson Education, 1998.
3. Alan Cooper, *The Essential Of User Interface Design*, Wiley – Dream Tech Ltd., 2002.
4. Russ Unger, Carolyn Chandler, *A Project Guide to UX Design: For user experience designers in the field or in the making*, 2012
5. Jeff Gothelf, *Lean UX: Applying Lean Principles to Improve User Experience*, 2016.
6. Wikipedia, Bart Pursel, *Information, People, and Technology, Creative Commons Attribution-ShareAlike 4.0 International License (Chapter 11)*

E28 DIRECTED STUDY

Elective **Hours: 4** **Credits: 3**

ALLIED COMPUTER SCIENCE I (INTRODUCTION TO PROGRAMMING) – UG
OBJECTIVES:

- An exposure to Computer Programming.
- To understand Basics of Python Programming.
- To learn Object Oriented Programming for beginners.
- Hands on experience with problem solving using Python

Unit 1
Introduction: Python, features of Python, documentation, program output, print statement, input, comment, variable, expression, statement, assignment. Function - calling functions, creating functions, formal arguments, and local variables.

Unit 2
Conditionals: Logical operators, conditional execution, alternative execution, chained conditionals and nested conditionals. Code blocks, indentation
Iterators: Multiple assignment, while statement, for statement.
Fruitful Functions and Recursion.

Unit 3
Strings, Lists and Tuples: Accessing elements, operators, built-in functions.

Unit 4
Dictionaries. Introduction to Dictionaries. Operators. Built-in Functions. Built-in Methods. Dictionary Keys.
Files: Open(), Read(), Write(), Close(). Directories.

Unit 5
Exceptions in Python. Detecting and Handling Exceptions. Exceptions as Strings. Standard Exceptions.

OUTCOMES:
On completion of the course the learner wil be able to

- Familiarize with Coding and Problem solving.
- Gain knowledge in Computer bascis.

References

1. Allen Downey, Jeffrey Elkner and Chris Meyers, *How to think like a Computer Scientist Learning with Python*, Second Edition, Soho Books, 2009
2. Sedgewick and Wayne,Dondero, *Introduction to Programming in Python*, Third Edition Pearson Education, 2016
3. Mark Lutz, *Learning Python : Powerful Object Oriented Programming*, Fourth Edition, O,Reilly Media, 2013

ALLIED COMPUTER SCIENCE II (OBJECT ORIENTED PROGRAMMING) - UG

OBJECTIVES:

- To teach OO concepts like abstraction, encapsulation, information hiding, inheritance and polymorphism with examples
- To help the students to develop programming skills and to solve problems using the various OOP Concepts.

Unit 1

Fundamentals of Object Oriented Programming: abstraction, encapsulation, information hiding, inheritance and polymorphism. object-oriented programming languages. Basics of C++: variables, constants, operators and expressions; Control statements – if statements, for loop, while loop, using continue, using break;

Unit 2

Arrays: one-dimensional and two-dimensional arrays; Functions in c++: call by value and call by reference, arguments, return types, scope, default arguments.

Unit 3

Object-oriented features of C++: classes and objects; member functions, messages, Private member functions, Arrays with in a class, friendly functions, constructors, destructors, copy constructors, access control specifiers,

Unit 4

Overloading: overloading member function, overloading constructor; operator overloading - unary and binary operators

Inheritance: simple - multilevel - multiple;.

Unit 5

Virtual base classes, Abstract classes; overriding member functions; pointers, virtual functions and **Polymorphism. Exception handling**.

OUTCOMES:

- Understand the various OO concepts.
- Write and execute programs in C++ for all the concepts
- Learn to use effective algorithms in problem solving

References

1. Bjarne Stroustrup, *The C++ Programming Language,* .Fourth Edition, Addison Wesley, 2013
2. Horton, Ivor, *Beginning C++*, Aprèss, 2014
3. Herbert Schildt, *C++: The Complete Reference*, Fifth Edition, McGraw Hill, 2012
4. E.Balagurusamy, *Object-oriented programming with C++*, Tata McGraw Hill, Seventh Edition, 2017
5. w3schools.com/cpp/
6. tutorialspoint.com/cplusplus/index.htm