SACRED HEART COLLEGE (AUTONOMOUS), THEVARA

KOCHI, KERALA, 682013

CURRICULUM AND SYLLABI

CHOICE BASED COURSE CREDIT AND SEMESTER SYSTEM

(CBCSS)

INTEGRATED M SC PROGRAMME IN COMPUTER SCIENCE – DATA SCIENCE (5 YRS)

Introduced from 2021 Admission Onwards

BOARD OF STUDIES IN COMPUTER SCIENCE

Sacred Heart College (Autonomous), Thevara, Kochi, Kerala

Integrated M.Sc. Programme in Computer Science - Data Science

Syllabus and Curriculum of the Programme Semester 1 to 6

First Semester

Course Code	Course Title	Common/ Complementary/ Core	Hours/week	Credits
IC1	English	Common	5	4
IC1CMT01	Mathematics - I	Complementary	4	4
IC1CRT01	Programming in C Language	Core	4	3
IC1CRT02	Introduction to Computers	Core	3	3
IC1CRT03	Database Management Systems	Core	3	3
IC1CRP01	Software Lab I	Core	6	4
Total				21

Second Semester

Course Code	Course Title	Common/ Complementary/ Core	Hours/week	Credits
IC2	Second Language	Common	5	4
IC2CMT02	Mathematics - II	Complementary	4	4
IC2CRT04	Object Oriented Programming using CPP	Core	3	3
IC2CRT05	Data Structures	Core	3	3
IC2CRT06	Operating Systems	Core	4	4
IC2CRP02	Software Lab II	Core	6	4
Total				22

Third Semester

Course Code	Course Title	Common/ Complementary/ Core	Hours/week	Credits
IC3CET01	Digital Electronics	Complementary	4	4
IC3CRT07	Advanced Computation Techniques	Core	4	4
IC3CRT08	Programming in Python	Core	3	3
IC3CRT09	R Programming and Mathematics for Artificial Intelligence	Core	4	3
IC3CRT10	Computer Organization and Architecture	Core	4	4
IC3CRP03	Software Lab III	Core	6*	2
Total				
*3 HRS FOR PYTHON AND 3 HRS FOR R PROGRAMMING.(External Lab exam should include both the topics)				20

Fourth Semester

Course Code	Course Title	Common/ Complementary/ Core	Hours/week	Credits
IC3	English - II	Common	4	4
IC4CET02	Micro Processors	Complementary	4	3
IC4CRT11	Data Mining	Core	4	4
IC4CRT12	Software Engineering	Core	4	3
IC4CRT13	Basics of Artificial Intelligence	Core	4	3
IC4CEP01	COMPLEMENTARY LAB	Complementary	2	2
IC4CRP04	Software Lab IV	Core	3	1
Total				20

Fifth Semester

Course Code	Course Title	Common/ Complementary/ Core	Hours/week	Credits
IC5CRT14	Principles of Machine Learning	Core	3	4
IC5CRT15	Web Application Development Using PHP	Core	4	3
IC5CRT16	Programming in Java	Core	4	3

IC5CRT17	IT and Environment	Core	3	4
	Project – Phase I	Core	3	
IC5CRP05	Software Lab V	Core	8*	3
Total			25	17
*3 HRS LAB FOR JAVA, 3 HRS FOR WEB APPLICATION AND 2 HRS FOR MACHINE LEARNING.(External lab exams needs to include all these topics and should have equal weightage)				

Sixth Semester

Course Code	Course Title	Common/ Complementary/ Core	Hours/week	Credits
IC6CRT18	Linux and Shell Programming	Core	3	3
IC6CRT19	Neural Networks and Deep Learning	Core	4	4
IC6CRT20	Computer Networks	Core	4	4
IC6CRT21	Mobile Application Development using Kotlin	Core	3	3
IC6CPP01	Project – Phase II	Core	7	4
IC6CRP06	Software Lab VI	Core	4*	2
TOTAL			25	20
*2 RS FOR LINUX, 2 HRS FOR MOBILE APPLICATION DEVELOPMENT (External Lab Examination should cover both these topics)				

TOTAL CREDITS FROM FIRST SEMESTER TO SIXTH SEMESTER IS 120

Semester I

Paper 1: Common – English – As approved by Bos of English (UG)

Number of hours / week: 5
Total number of Instructional Hours: 90
Credits: 4

Fine Tune Your English

Division 1

1. How to Write Effective Sentences
2. Phrases – What are They? -

3. The Noun Clauses –
4. The Adverb Clause
5. If All the Trees Were Bread and Cheese
6. The Relative Clause
7. How the Clauses Are Conjoined

Division 2

8. Understanding the Verb
9. Understanding the Auxiliary Verb
10. Understanding the Adverbs
11. Understanding the Pronoun
12. The Reflexive Pronoun

Division 3

13. The Articles I
14. The Articles II
15. The Adjective
16. Phrasal Verbs
17. Mind Your Prepositions
20. Errors, Common and Uncommon
21. False Witnesses

Division 4

47. 'Presentness' and Present Tenses
48. The Presentness' of a Past Action
49. Past Tense
50. Futurity in English
51. Passivization
52. Animal Expressions
53. Idiomatic Phrases
18. Concord

Division 5

56. Negatives
57. How to Frame Questions?
58. What's What?
59. The Question Tag
63. Is John there, please?
32. Word Information

33. Use the specific word
37. Body vocabulary
74. Letter Writing

Paper 2: Complementary - Mathematics - I – As approved by Bos of Mathematics (UG)

Number of hours / week: 4
Total number of Instructional Hours: 72
Credits: 4

Graph Theory and Linear Programming

Module I: Graphs (18 hrs)

Graphs and Graph Models, Graph Terminology and Special types of Graphs, Representing Graphs and Graph Isomorphism, Connectivity, Euler and Hamilton Paths.
Text 1 Chapter 8 (Sections 8.1, 8.2, 8.3, 8.4 and 8.5 only)

Module II:Trees (18 hrs)

Introduction to Trees, Application of Trees, Tree Traversal, and Spanning Trees.
Text 1 Chapter 9 (Sections 9.1, 9.2, 9.3 and 9.4 only)

Module III: Linear programming problems (18 hrs)

Mathematical formulation of a L.P.P., General linear programming problems, solution of a L.P.P, graphical method for solving a L.P.P.

Simplex Method: Stack and surplus variables- reduction of any feasible solution to a basic feasible solution. Unbounded solution. Optimality conditions- artificial variable techniques- Big M method.

Module IV: Transportation & Assignment Problems (18 hrs)

Transportation model- solution by simplex method- north west corner rule, lowest cost entry method,Vogel method, MODI method, degeneracy, assignment problems.

Text Books

1. **Kenneth H Rosen ; Discrete Mathematics And Its Applications ; 6th Edition ; Tata Mc Graw-Hill Publishing Company Limited**

2. **Belly E Gillet – Introduction to Operations Research (A Computer Oriented Arithmetic Approach) (Tata Mc. GrawHill)**

Paper 3: Core – Programming in C Language

Number of Hours / Week: 4
Total Number of Instructional Hours: 72
Credits: 3

Course Objective: On completion of the course, the student will be able to write a complete C program, he / she will be able to use decision making statements and looping structures, should have a clear concept on one-dimensional, two-dimensional arrays, modular programming using user defined functions, clarity on concept of strings, structures and unions. Should be able to use files for input and output, basic ideas on dynamic storage allocation and command line arguments.

Unit 1: (14 Hours)

Algorithm and flow chart (structure, desirable qualities, simple examples (sequential, branching, and iterative)). Basic concepts in machine language program and program execution, assembly language program and assembler, high level language program and translators (compilers and interpreters). Procedural programming paradigm – examples. Steps in developing a program – (problem analysis algorithm design, coding, debugging, testing, documentation). Approaches – top down and bottom up approaches. C Language – structure of a C program – simple dequential program. Role of editor. Compilation, linking and execution under Windows and Linux. IDEs. Types of errors.

Unit 2: (12 Hours)

Keywords, constants, variables, data types and variable names, assignment statement. Operators and expressions (including increment, decrement and sizeof()), precedence and order of execution, mixed mode expressions and type conversions. Elementary ideas in function (C program as a collection of functions), main function. Formatted input and output. Simple sequential programs.

Decision making: The goto statement, if, if-else, nesting of if, else if ladder and switch statement, conditional expression. Example programs based on decision making.

Unit 3: (14 Hours)

Control Statements: The while loop, the do..while loop, the for loop, nesting of loops, the break statement and continue statement. Example programs.

Functions, basics, prototype, parameter passing, storage classes, recursion. Built-in functions. Example programs.

Unit 4: (16 Hours)

Arrays, arrays and functions, Strings, string operations and algorithms, string functions in C. example programs using arrays and strings (including simple search and sort, matrix operations).

Pointers – basic concepts, pointer arithmetic, pointers and arrays, pointers and strings, pointers and functions. Dynamic memory allocation, simple programs using pointers.

Unit 5: (16 Hours)

Structures – basics, array of structures, pointers and structures, structure and function, self – referential structures, union, programs using structures.

The Pre-processor: File Inclusion, Macro definition and substitution, macros with arguments, nesting of macros, conditional compilation. (Simple illustrative examples)

File Management: Defining and opening a file, closing files, input/output operations on files, predefined streams, error handling during I/O operations, Random Access to files, Command line arguments. Simple examples of file creation.

Book of Study: Programming in ANSI C: E Balagurusami 8th edition MC GRAW HILL INDIA publishers.

Reference texts:

1. Programming with C: Byron S Gottfried, Schaumes Outline series, 4th edition.
2. Programming in C: Ashok n Kamthane, Pearson Education, 3rd edition
3. Let us C: Yeshwant Kanetkar 16th edition BPB publishers

Paper 4: Core – Introduction to Computers

Number of Hours / Week: 3
Total Number of Instructional Hours: 54
Credits: 3

Course Objective: After successful completion of the course, the students will be able to –

- **Understand basic functions of computer hardware, software components including memory and operating systems**
- **Understand the concept of networking and internet**
- **Understand IT and its impact on society**

Unit 1 (8 Hrs)

Introduction: Functional units of a computer system, Different types of computers, Computer Software and Hardware, Types of Software (System software and Application software). Characteristics of Computers. Computer Languages (Machine, Assembly, and High Level Languages – 3GL, 4GL, 5GL)

Unit 2 (12 Hrs)

Interaction with Computers, Data Processing and Storage Information: Input devices, Output devices, Representation of Data, Processing of Data, the CPU, Memory, different types of RAM and ROM, types of storage devices (Magnetic storage devices, Optical storage devices, Solid State storage devices), SSD – types, performance, benefits; Graphics Processing Unit (GPU)

Unit 3 (12 Hrs)

Introduction to Operating Systems, Networking: Definition of an Operating System, Different types of PC Operating systems, File Management (file access methods, file operations, file naming). Computer Network: Basic elements of communication system, Data transmission modes, data transmission speed, data transmission media (twisted pair wire, coaxial cable, Microwave system, Communication satellite, Optical fibres), Modems, Categories of networks (PAN, CAN, LAN, WAN, MAN).

Unit 4: (12 Hrs)

Internet: Definition, Working of internet, Major features of internet, Major services (WWW, Electronic mail, FTP, Chat, Instant messaging, Telnet, Usenet News, Online Services, Peer-to-Peer services), TCP/IP, URL's, Web Browsers, Major elements of Internet Search engines, Popular Search Engines, Uses of the Internet, WWAN, Academic service (INFLIBNET, NPTEL, NICNET, BRNET)

Unit 5: (10 Hrs)

Introduction to Cyber World: Cyber space, introducing cyber laws, scope of cyber laws (E-commerce, online, contracts, IPRs (copyrights, trademarks, and software patenting); cyber ethics, Cyber addiction – types of internet addiction, causes of cyber addiction, effects of Internet addiction, effects of internet addiction; Cyber crimes – Introduction, categories of cyber crime, types of cyber crimes.

Books of Study:

1. Peter Norton's – Introduction to Computers, Sixth edition, Published by Tata McGraw Hill
2. P.K Sinha & Priti Sinha – Computer Fundamentals, Fourth edition, BPB Publications.
3. Barkha & U. Rama Mohan – Cyber Law Crimes, Asia Law House, New Edition

Refernce texts:

1. V Rajaraman – Introduction to Information Technology, Prentice-Hall of India.
2. Harley Hahn – The Internet Complete Reference, Tata McGraw Hill edition
3. Dr. Farooq Ahamad – Cyber Law in India (Law of Internet), New Era Law publication.

Paper 5: Core – Database Management Systems

Number of Hours / Week: 3
Total Number of Instructional Hours: 54
Credits: 3

Course Objective: On completion of the course, the student should have a clear concept on databases, data models, architecture and components of DBMS. The concept of entity, attributes, associations and relationships, concept of tables and its properties, table creation and manipulation of tables and databases using SQL. DDL and DML facilities of SQL.

Unit 1: Database Management System Concepts (9 Hrs)

Introduction, Significance of Databases, Advantages of Database approach; Data Independence; Components of Database Systems, Classification of Users, the Database Administrator (DBA) and his responsibilities; advantages and disadvantages of Database Management System.

Unit 2: Entity attributes and Data Models for a Database (9 Hrs)

Entities and their Attributes, different types of Entities and Attributes, Association and relationships and their different types. E-R Diagrams: Data Models, Hierarchical, Network and Relational data models. Benefits and Application of each Data Model.

Unit 3: DBMS Architecture and Schema, Data Dictionary (10 Hrs)

Three Level Architecture of DBMS, The External Level or Subschema, The Conceptual Level or Conceptual Schema, The Internal Level or Physical Schema, Data Definition Language, Data Manipulation Language; Database Management System Structure, Database Manager, Database

Administrator, Data Dictionary; Brief Introduction to Distributed Databases and Client / Server Architecture

Unit 4: The Relational Approach to DBMS (12 Hrs)

The Concept of relations in Mathematics, Mathematical concepts of sets, relations and functions, Relational approach to DBMS – Attributes and Domains; concept and properties of tables, cardinality and degree of relations, keys and different types of keys; strong entities and weak entities. Entity integrity rule, the foreign key and the rule of referential integrity. Representation of relational database schemas, integrity constraints and different types; Relational Algebra. Operators in Relational Algebra.

Unit 5: The Structured Query Language (SQL) (14 Hrs)

The need for SQL. Brief Introduction to query language and its evolution. Basic structure of SQL queries, Data Definition Commands: Data Types in SQL, CREATE, ALTER, DROP commands, Adding constraints in SQL, Basic operations in Data Manipulation using SQL, INSERT, SELECT, DELETE, UPDATE, Substring comparison using LIKE operator, BETWEEN operator, SQL set operations UNION, EXCEPT, INTERSECT. Order By and Group By clauses, complex queries in SQL, Nested queries, EXISTS and UNIQUE functions, Renaming of attributes and joining of tables, Aggregate functions, Creating and Managing Views.

Book of Study: Raghu Ramakrishnan & Johannes Gehrke, Database Management Systems, McGraw Hill International Edition

Reference Texts:

1. Fundamentals of Database Systems 7th Edition, Elmasri & Navathe, Pearson Education
2. An introduction to database systems, C. J Date, 8th edition, Pearson education
3. Abraham Silberschatz, Henry F Korth & S Sudharssan, Database System Concepts, 4th edition, Tata McGraw Hill.

Paper 6: Core – Software Lab I

Number of Hours / Week: 6
Credits: 4

The candidate needs to submit a Lab Record, duly signed by the teacher in charge and Head of the Department, (minimum of 20 programs, (10 + 10, C and SQL), failing to which he / she will not be allowed to attend the external software lab examination. The Lab record should be hard binded

with name of the college and the emblem of the college depicted on the first page and should be properly indexed.

Syllabus for C Programs

1. Simple programs to familiarize printf() and scanf() functions.
2. Programs based on decision making statements, break, goto, continue, switch.
3. Programs using Loop controls statements.
4. Programs based on one dimensional and two-dimensional arrays (linear search, sort, matrix addition, multiplication, transpose etc.)
5. Programs on strings and string handling functions
6. Programs using the concept of pointers, operations on pointers, pointers to one dimensional array
7. Programs using the concept of functions, call by value, call by reference, recursion.
8. Programs based on structure and union, array of structures, pointer to structure, structure as argument to functions.
9. Simple programs using pointers and malloc().

Syllabus for SQL Programs

Problems involving the following topics to be included

1. Data definition commands – CREATE, ALTER, DROP, Adding Constraints, Primary Key, Foreign Key, Unique Key, check, not null.
2. Basic SQL queries INSERT, SELECT, DELETE, UPDATE, Using multiple tables, ordering of rows using ORDER BY option, Set operations using UNION, EXCEPT, INTERSECT, Substring comparison using LIKE operator, BETWEEN operator.
3. Complex Queries: Netsed Queries, EXISTS and UNIQUE / DISTINCT functions, NULL values, Renaming of attributes and Joining of tables Aggregate functions and grouping.
4. Managing views, Simple stored procedures

Semester II

Paper 1: Common – Syllabus of Second Language: As approved by the respective BoS (UG)

Paper 2: Complementary – Syllabus of Complementary paper – Mathematics: As approved by BOS of Mathematics (UG)

Linear Algebra

Number of Hours / Week: 4
Total Number of Instructional Hours: 72
Credits: 4

Module I: Introduction To Vector Spaces **15 Hrs.**

Vector Spaces: Rn and Cn, lists, Fnand digression on Fields, Definition of Vector spaces, Subspaces, sums of Subspaces, Direct Sums, Span and Linear Independence, bases, dimension.

Module II: Linear Maps **20 Hrs.**

Definition of Linear Maps - Algebraic Operations on - Null spaces and Injectivity - Range and Surjectivity - Fundamental Theorems of Linear Maps - Representing a Linear Map by a Matrix - Invertible Linear Maps - Isomorphic Vector spaces - Linear Map as Matrix Multiplication - Operators - Products of Vector Spaces - Product of Direct Sum - Quotients of Vector spaces.

Module III: Eigenvalues, Eigenvectors and Eigenspaces **20 Hrs.**

Eigenvalues and Eigenvectors - Eigenvectors and Upper Triangular matrices - Eigenspaces and Diagonal Matrices.

Module IV: Inner Products and Norms **17 Hrs.**

Inner Products, Norms, Orthonormal Bases, Self Adjoint and Normal Operators, Spectal theorem, Polar Decomposition and Singular Value Decomposition.(proof of all theorems are excluded for module 4)

Text : Sheldon Axler, Linear Algebra Done Right, Third Edition Springer, 2017.

Paper 3: Core – Object Oriented Programming using C++

Number of Hours / Week: 3
Total Number of Instructional Hours: 54
Credits: 3

Course Objective: On completion of the course, the student will be able to understand

1. **Object Oriented programming concepts and introduction of C++ Programming language.**
2. **Different control structures used in C++ and implementation of functions in C++.**
3. **Importance of class and objects concept in programming**
4. **Role of constructors and destructors and importance of operator overloading**
5. **Different types of inheritance and implementation of polymorphism**

Unit 1: (10 Hrs) – Principles of Object-Oriented Programming, Beginning with C++

Object Oriented Technology, Disadvantages of conventional programming, Programming Paradigms, Key concepts of Object oriented programming, Advantages of OOPs, Parts of C++ program, Types of tokens, Data types in C++, Type modifiers, Type casting, Constants, Constant pointers, Operators in C++, Referencing and dereferencing operators, Scope access operator, Memory management operators.

Unit 2: (12 Hrs) – Control structures and functions in C++

Decision making statements, Loops in C++, Functions in C++, The main function, Parts of a function, Passing arguments, Return by reference, Default arguments, Inline function, Function overloading, Principles of function overloading.

Unit 3: (10 Hrs) – Classes and Objects

Structures in C++, Class in C++, Declaring Objects, public private protected keywords, Defining member functions, Characteristics of member functions, Outside member function inline, Rules for inline functions, Data hiding, Memory allocation for objects, Static member variables and functions, Arrays of objects, Objects as function arguments, Friend functions, Friend functions, Friend classes, Local classes

Unit 4: (10 Hrs) – Constructors, Destructors and Operator Overloading

Constructors and destructors, Characteristics of constructors and destructors, Applications with constructors, Constructors with arguments, Overloading Constructors, Constructors with default arguments, Copy constructors, Destructors, Operator overloading, Overloading of unary operators, Overloading binary operators, Overloading with friend functions, Type conversion, Rules for overloading operators.

Unit 5: (12 Hrs) – Inheritance, Pointers, Binding, Polymorphism and Virtual functions

Inheritance, Access specifiers and simple inheritance, Types of inheritance, Virtual base classes, Constructors, Destructors and Inheritance. Pointer, Pointer declaration, void pointers, wild pointers, Pointers to objects, this pointer, pointers to derived classes and base classes, Binding in C++, Pointers to derived class objects, Virtual functions, Rules for virtual functions, Pure virtual functions, Abstract classes, Working of virtual functions, Object slicing

Book of Study:

1. Ashok N. Kamthane Object Oriented Programming with ANSI & Turbo C++, First Edition, Pearson India.

Reference:

1. E. Balagurusamy – Object Oriented Programming with C++, Fifth edition, Tata McGraw Education Hill, 2011.
2. Ravichandran – Object Oriented Programming in C++, TMH, 3rd Edition

Paper 4: Core – Data Structures Using C++

Number of Hours / Week: 3
Total Number of Instructional Hours: 54
Credits: 3

Course Outcomes: Upon successful completion of this course, students should be able to:

1. **Describe fundamental concepts of data structures.**
2. **Illustrate the representation of arrays in memory and operations on it.**
3. **Compare and contrast different searching and sorting techniques.**
4. **Design operations on linear data structures such as stacks and queues.**
5. **Implement operations on various types of linked lists.**

Unit 1: Introduction to Data structures (10 Hours)

Introduction to Data Structures, Definition, Classification of Data Structures, Primitive and Non primitive, Operations on data structures, Static and dynamic memory allocation. Dynamic memory allocation and pointers, Memory allocation operators in C++. User defined data types in C++. Recursion, Recursive functions in C++.

Unit 2: Array (10 Hours)

Arrays, Linear array – Representation of array in memory, operations on linear array – Insertion, Deletion, Sorting and Searching. Two Dimensional Arrays – Representation of 2D array in memory, operations on 2D array, Multidimensional Arrays.

Unit 3: Search and Sort (10 Hours)

Search, Basic search techniques, Search algorithms, Searching techniques, Sequential search, Binary search. Sort, general background, definition, different types, Bubble Sort, Selection Sort, Merge Sort.

Unit 4: Stack and Queue (12 Hours)

Stack, Definition, Array representation of stack, Operations on stack. Infix, prefix and postfix notations, Conversion of an arithmetic expression from infix to postfix, Postfix evaluation, Applications of stack. Queue – Definition, Array representation of queue, Simple queue operations. Circular queues, Double ended queue, Priority queue.

Unit 5: Linked List (12 Hours)

Linked list – definition, Components of linked list, Representation of linked list, Advantages and disadvantages of linked lists, Types of linked list. Singly linked list, Operations on singly linked list, Creation, Insertion, Deletion, Search and display. Circular linked list, Operations on circular linked list, Creation, Insertion, Deletion, Search and Display.

Book of Study: G.S Baluja, Data Structures Through C++ (A Practical Approach), Danapat Rai & Co.

References:

1. Ellis Horowitz and Sartaj Sajni, Fundamentals of Data Structures, Galgotia publications
2. Seymour Lipschutz, Theory and Problems of Data Structures, Schaums Outline Series

Paper 5: Core – Operating System Concepts

Number of Hours / Week: 4
Total Number of Instructional Hours: 72
Credits: 4

Course Outcomes:

After completing the course, the student should be able to explain

1. **The fundamental concepts regarding an OS**

2. **Concept of a process and management of processes**
3. **Inter process synchronization methods and deadlock handling**
4. **Various memory management techniques**
5. **Concept of file and various file handling methods**

Unit 1: Introduction (14 Hrs)

OS Definition, Functions, Types of operating systems – Batch Operating System, Multi programming, Time sharing, Real time, distributed operating systems – Operating System Operations, Operating System Services, User Operating System Interface, System Calls, Types of System Calls.

Unit 2: Processor Management (14 Hrs)

Job and process concept, Operating system view of process, process state, state transition diagram, PCB (Process control block), System state and process lists, process switch, threads, multi-threading operating system, operating system services for process management. Process Scheduling – Types of schedulers, scheduling and performance criteria, scheduling algorithms, multiple processor scheduling.

Unit 3: Inter process synchronization and communication (16 Hrs)

Concurrent Processes, need for inter process synchronization, critical section problem, mutual exclusion, mutual exclusion algorithms, semaphore definition, primitives, implementation of semaphores, monitors

Deadlocks – Definition, Deadlock characterization, Resource allocation graph, methods for handling deadlocks, deadlock prevention, deadlock avoidance, safe state, resource allocation graph algorithm, Banker's algorithm, deadlock detection, recovery from deadlock.

Unit 4: Memory Management (16 Hrs)

Preliminaries, address binding, dynamic linking and loading, Overlays. Logical versus physical address space, Swapping, Contiguous allocation – fragmentation, compaction, Paging – principles of page allocation, structure of page table, hardware support, multi level paging, Segmentation – principles of operation, hardware, implementation of segment table, protection and sharing, fragmentation, segmentation with paging. Virtual Memory – Demand paging, Page replacement algorithms, page allocation policies – Thrashing, hierarchical address translation tables, MMUS

Unit 5: File Management (12 Hrs)

File Structure, File types, File access, File attributes, File operations, Directories – Flat directory systems, hierarchical directory systems. File system implementation – Allocation methods, contiguous allocation, linked allocation, indexed allocation.

Book of Study

1. Siberschatz, Galvin, Gagne: Operating System Concepts, 7th Edition

Reference Texts:

1. Andrew S. Tanenbaum, Modern Operating Systems, Prentice Hall India
2. Dhamdhere, System software and operating systems – Tata Mc Graw Hill
3. H M Deitel, An introduction to Operating System – Adison Wesley
4. Tanenbaum, Modern Operating Systems – Prentice Hall
5. William Stallings, Operating Systems – Pearson Education

Paper 6: Core – Software Lab II

Number of Hours / Week: 6
Credits: 4

The candidate needs to submit a Lab Record, duly signed by the teacher in charge and Head of the Department, (minimum of 20 Programs, (10 + 10, CPP and DS) failing to which he / she will not be allowed to attend the external software lab examination. The lab record should be hard binded with the name of college and the emblem of the college depicted on the first page and should be properly indexed.

Syllabus for CPP programs (minimum of 10 questions)

1. Programs based on array of objects, friend functions, passing objects as arguments to function.
2. Programs based on array of objects, friend functions, passing objects as arguments to function.
3. Programs based on operator overloading (binary, unary) using member functions and friend functions
4. Programs based on constructors, different types of constructors.
5. Programs based on inheritance, different types of inheritance, polymorphism

Syllabus for Data Structures using CPP (minimum of 10 questions)

Student needs to code and implement CPP programs for the following:

Arrays – Insertion, Deletion, Polynomial addition using arrays
Sort – Selection, Insertion, Quick

Search – Linear search, Binary search

Sparse matrix –

Sparse form representation, transpose and addition using the sparse form

Stack –

Implementation using arrays (linear stack), Infix to postfix conversion, Postfix evaluation

Queue –

Implementation using arrays (linear queue), Implementation of circular queue

Singly linked list – Implementation using dynamic memory allocation techniques, arrange the list based on the ascending or descending order of the information field, concatenate two linked lists, interchange any two nodes in a list, Implementation of circular list, Implementation of linked stacks and queues.

Doubly linked list – Implementation of doubly linked list, Implementation of circular doubly linked list.

Semester III

Paper 1: Complementary Course in Electronics

DIGITAL ELECTRONICS

(*Submitted as suggested by the BOS of Electronics (PG and UG) of Mahatma Gandhi University*)

Course Outcome

After completion of the course the students will be able to

CO1: Explain number systems, describe binary and hexadecimal arithmetic and Discuss on logic gates.

CO2: Analyse Boolean expressions using logic gates, Discuss Boolean algebra and Minimize SOP and POS expressions using Karnaugh map.

CO3: Illustrate and explain various combinational logic circuits

CO4: Explain various sequential logic circuits, Design various Up and down counters and describe various shift registers and shift register counters.

(*PSO that can be linked to: Understand the core concept of computer science for applying and developing logic circuits, understanding processors and assembly language programing*)

(SYLLABUS)

Module I **14 hours**

Review of number systems: Decimal Numbers, Binary numbers, Decimal to binary and binary to decimal conversions, Conversion of hexadecimal numbers to binary and decimal

number system, Introduction to digital electronics: digital and analog systems, binarydigits, logic levels, Binary Arithmetic, 1's and 2's complement of binary numbers, Signed numbers, Arithmetic operations with signed numbers, Arithmetic operation of hexadecimal numbers (addition and subtraction) Binary coded decimal, 8421 BCD code, BCD addition, Digital codes: Gray code, ASCII and EBCDIC codes.

Chapter 2 (Pages 16 to 47 and 51 to 60) of Text book 1

Module II **12 hours**

Logic Gates: AND, OR, NOT, XOR, XNOR, NAND (Definition, Symbols, Truth Tables and Operation, gate propagation delay time, fan-in and fan-out). Basic combinational logic circuits, Implementing combinational logic, Universal property of NAND and NOR gates, Combinational logic using NAND and NOR gates, Realization of basic gates using NAND and NOR gates. *Chapter 3 (Pages 78 to 103) and Chapter 5 Pages 172 to 185 of Text book 1*

Module III **15 hours**

Boolean algebra Boolean variables, operations and functions, Boolean Postulates, Boolean Theorems, De-Morgan's theorems, Boolean analysis of logic circuits, simplification using Boolean algebra, standard forms of Boolean expression, Boolean expressions and truth tables. The Karnaugh Map, Karnaugh SOP and POS minimization. *Chapter 4 (Pages 4.5 to 4.42) of Text book 2*

Module IV **16 hours**

Basic overview of logic functions, Introduction to combinational circuits: Basic adders (adders and subtractors), parallel binary adders, ripple carry and look ahead carry adders, magnitude comparators, decoders, encoders, code converters: binary to Gray and Gray to binary converters, multiplexers (Data selectors), De-multiplexer, parity generators/ checkers.

Introduction to sequential circuits: Latches, flip flops: RS flip flop using NAND/ NOR gates, Clocked RS, D, JK and T flip flops, Edge triggered flip flops, Master-slave JK flip flop, basic flip flop applications *Chapter 6 (Pages 202 to 251) and Chapter 7 (Pages 266 to 287)of Text book 1*

Module V **15 hours**

Basic shift register functions. Serial in- Parallel out shift registers, Parallel in -Serial out shift registers, Serial in- Serial out shift registers, Parallel in Parallel out shift registers, Bidirectional shift registers Shift register counters.

Types of counters- asynchronous and synchronous counters, design of asynchronous and synchronous counters, up/down counters, decade counters, cascaded counters, applications of counters. *Chapter 8 (Pages 305 to 336) and Chapter 9 (Pages 354 to 374) of Text book 1*

Text Books:

1. Digital Fundamentals- Floyd and Jain, Eighth Edition
2. B. RAM, "Computer Fundamentals: Architecture and Organization", FourthEdition, New age international (P) Limited.

Paper 2 Core: IC3CRTO7 Advanced Computation Techniques

Instructional Hours/Week: 4
Total Instructional Hours: 72
Credits: **4**

Upon successful completion of this course, students should be able to:

1. Illustrate the representation of tree structure and basic operations on it
2. Describe fundamental concepts of File as a data structures.
3. Gain the needed expertise in designing normalized databases
4. Attain fundamental concepts of concurrent transactions and Database security
5. Understand the storage techniques and accessing data from databases

Unit 1: Trees: Introduction, Tree Terminology, Binary trees-Strictly Binary Tree, Complete Binary Tree, Extended Binary Tree, Binary Tree Representation- Array and Linked List representation, creation of Binary Tree, Operations on Binary Trees, Traversal of Binary Tree- Inorder, Preorder and Post order traversals, Technique of Conversion of an expression into Binary Tree, Binary Search Tree- Insertion of a node, Searching for Node, Deletion of a Node. (16 HRS)

Unit 2: Files: Introduction, Terminology, File Organization, File Operations- Creation of a Fie, Reading of a File, Updation of a File, Insertion in the file, Deletion from the File, Sequential Files-Structure, Operations, Disadvantages, Areas of Use, Indexed Sequential File- Structure of Indexed Sequential File, Direct File Organization, Multiple Key Access- Multilist Organization, Inverted-List Organization. (16 HRS)

Unit 3: Relational Database design: features of good relational database design. Un-normalized relations, insertion deletion updation anomalies, normalized relations, first, second , third and BCNF normal forms, functional dependency, non-loss decomposition of relations, Concept of multivalued dependency, Basic concept of object oriented DBMS, databases connectivity (16 HRS)

Unit 4: Transaction Management and Concurrent transactions: properties of transactions, benefits of concurrent transactions, schedules and locking principles, problems and solution of concurrent transactions, Database security and authorizations , Grant and Revoke , triggers and its need, system crashes and recovery, concept of logs (12 HRS)

Unit 5: Overview of Storage and Indexing: Data on External Storage, storage of databases and access methods, Indexing and data retrieval, types of indexes - Clustered Indexes, Primary and Secondary Indexes, Index data Structures - Hash Based Indexing, Tree based Indexing, Comparison of File Organizations. (12 HRS)

Text Books:

1. G.S Baluja, Data Structures through C++ (A Practical Approach), Danapat Rai & Co.
2. Fundamentals of Database systems 7th edition: Elmasri & Navathe : Pearson Education.
3. Database Management Systems by Raghu Ramakrishnan, Johannes Gehrke 3rd. Edition, July 2014, McGraw Hill Education.

References:

1. Ellis Horowitz and Sartaj Sahni, Fundamentals of Data Structures, Galgotia publications.
2. An introduction to database systems C.J Date 8th edition. Pearson education

Paper 3 Core: IC3CRT08 Programming in Python

Instructional Hours/Week: 3
Total Instructional Hours: 54
Credits: **3**

Aim: To provide basic knowledge in Python programming

Course objectives: After completing this course, the student will learn how to design and program Python applications, acquire object oriented skills in Python and able to work with python standard library.

Module I

Programming Environment and Python Basics: Getting Started with Python Programming - Running code in the interactive shell, Editing, Saving, and Running a script. Using editors - IDLE, Jupyter. Basic coding skills – Writing simple programs. (10 HRS)

Module II

Building Python Programs: Data types, variables, operators. Control statements – branching controls, simple if, if - else, if - elif -else; looping, while, for. Functions - defining, calling,

returning values, functions with default arguments, recursive functions, nested functions and lamda functions. Strings - operation, string functions. Work with dates and times. (10 HRS)

Module III

Containers: Lists - Basic list Operations and functions, List of lists, Slicing, Searching and sorting list, List comprehension. Work with tuples and Sets. Dictionaries - Dictionary functions, dictionary literals, adding and removing keys, accessing and replacing values, traversing dictionaries, reverse lookup. (10 HRS)

Module IV

Object Oriented Programming: Design with classes, Inheritance – multi-level and multiple inheritance. Exceptions - Handle a single exception, handle multiple exceptions. Introduction to file I/O - Reading and writing text files, Manipulating binary files. More concepts: Decorators, generators and iterators. (12 HRS)

Module V

Scientific Python: NumPy - Basics, Creating arrays, Arithmetic, Slicing, Matrix Operations, Random Numbers.. Plotting and visualization. Matplotlib - Basic plot, Ticks, Labels, and Legends. Pandas: operations on CSV files. Reading, Manipulating, and Processing Data. Python GUIs and event handling using tkinter. (12 HRS)

Text Books:

1. Kenneth A Lambert., Fundamentals of Python: First Programs, 2/e, Cengage Publishing,2016
2. Jeeva Jose, P Sojan Lal, Introduction to Computing and Problem solving with Python, Khanna Book Publishing, 2016
3. Wes McKinney, Python for Data Analysis, 2/e, Shroff / O'Reilly Publishers, 2017

Reference Books:

1. Allen B. Downey, Think Python: How to Think Like a Computer Scientist, 2/e, Schroff, 2016
2. Michael Urban and Joel Murach, Python Programming, Shroff/Murach, 2016
3. David M.Baezly, Python Essential Reference. Addison-Wesley Professional; 4/e, 2009.

Paper 4 Core: IC3CRT09 R Programming and Mathematics for Artificial Intelligence

Instructional Hours/Week: 4
Total Instructional Hours: 72
Credits: **3**

Aim: To attain the needed proficiency in R programming and ability to develop programmes to solve Mathematical problems Using R

Course Outcome: On completion first two Units of the course, students will be able to use R language for programming purposes. The remaining three Units will enable the student to become confident in the Mathematical portions needed in the field of Artificial Intelligence. The Lab sessions for the Paper is so designed to make the student an expert in R to solve problems in Mathematics

Unit I

R Programming -Fundamentals, installation and use of software, data editing, use of R as a calculator, functions and assignments, arguments, scope, logic and statements in R, logical equivalence, Sets with R: Cardinality, Equality, Empty set, Subset, ,Union, Intersection, Complement, Cross product and Algebraic properties (12 hrs)

Unit II

R Programming - Exploring and cleaning data for analysis, Data organization, Arrays, and Matrices, Basics of Arrays in R, Matrix operations, Advanced Matrix operations, Additional Matrix facilities, Lists and Data frames. Mapping models to Machine Learning, Evaluating and Validating models, Probability distributions in R, Statistical models in R, Building, linear models, Generalized linear models, Nonlinear least squares and maximum likelihood models.(14 hrs)

Unit III

Sets, Operations on sets, Venn Diagrams, Multi Sets, Binary Relations, Equivalence Relations, Ordering Relations, Operations on Relations, Partial Orders. Statements and Notation, Connectives, Quantified Propositions, Logical Inferences, Methods of Proof of an Implication, First Order Logic and other Methods of Proof, Rules of Inference for Quantified Propositions, Proof by Mathematical Induction. (15 hrs)

Unit IV

Linear Algebra – System of Linear equations, Solving System of Linear equations, Linear Independence, Vectors, Scalars, Addition, Scalar multiplication, dot product, vector projection,

cosine similarity. Support Vector Machines, Implementation using Python, Classification using Support Vector Machines. (15 hrs)

Unit V

Matrices, determinants, inverse of matrix. System of equations, Linear transformation - rank and nullity, Consistency, and Inconsistency of linear system of equations, rank nullity theorem, Echelon form of a matrix and Row reduced echelon form of matrix. Correlation coefficient, Eigen values and Eigen vectors. Principle Component analysis (PCA) – Concepts and properties. Dimensionality reduction with PCA. (16 hrs)

Book of Study:

- N Matloaff ,"The art of R Programming", No Starch Press, Inc, 2011, Ist edition,ISBN-10: 1-59327-384-3, ISBN-13: 978-1-59327-384-2.
- William B. Claster, "Mathematics and Programming for Machine Learning with R: From the Ground Up" CRC Press; 1st edition (27 October 2020), ISBN: 9780367507855.
  For Maths
- Kenneth H. Rosen, "Discrete Mathematics And Its Applications", 7th Ed,McGrawHill, 2012.
- Erwin Kreyszig, "Advanced Engineering Mathematics", Wiley India, 9th Edition 2011.

Reference Books

- Nina Zumel, John Mount, Jeremy Howard, Rachel Thomas, "Practical Data Science With R", Manning Publications, Year: 2020, ISBN: 1617295876, 9781617295874.
- "Mathematics for Data Science and Machine Learning using R" by Eduonix, September 2019, Packt Publishing, ISBN: 9781839210945.
- Mark Gardener, "Beginning R: The Statistical Programming Language", ISBN: 978-1-118-16430-3 May 2012.

Web references

https://cran.r-project.org/doc/contrib/Verzani-SimpleR.pdf

Paper 5 Core: IC3CRT10 Computer Organization and Architecture

Instructional Hours/Week: 4
Total Instructional Hours: 72
Credits: **4**

Aim: To impart fundamental concepts about organization of a Computer system and its architecture to the students

Course objective: Upon successful completion of this course, students should be able to:

1. Describe the fundamental organisation of a computer system
2. Explain addressing modes, instruction formats and program control statements
3. Analyze the organization and performance of system memory hierarchy
4. Describe basic concept of parallel computing.
5. Describe fundamentals concepts of pipeline and vector processing

Unit 1:

Introduction: Functional units of a computer, Basic operational concepts, Bus structure, Memory locations and addresses, Instructions and instruction sequencing, Instruction execution**.** Instruction Formats, Addressing Modes (16 HRS)

Unit 2:

Central Processing Unit: General Register Organization, Stack Organization, Instruction Formats, Instruction Classification, Addressing modes. (14 HRS)

Unit 3:

Memory: Memory Hierarchy, RAM, ROM, Cache Memories, Virtual memory (12 HRS)

Unit 4:

Pipeline and Vector Processing: Parallel Processing, Architectural classification scheme-SISD, SIMD, MISD, MIMD, Pipelining, Arithmetic Pipeline, Instruction Pipeline, RISC Pipeline, Vector Processing, Array Processor. (16 HRS)

Unit 5:

Multi Processors:

Characteristics of Multiprocessors, Interconnection Structures, Interprocessor arbitration, Interprocessor communication and synchronization, Cache Coherence. (14 HRS)

Text Book:

1. Computer Organization, V. Carl Hamacher, Zvonko Vranesic, Safwat Zaky, 5th Edition, McGraw Hill Education.
2. Computer System Architecture, M. Morris Mano, Third Edition, Pearson/PHI.

References:

1. Computer Organization and Architecture – William Stallings, Sixth Edition, Pearson/PHI.

2. Computer Architecture and Parallel Processing, Kai Hwang and F. A. Briggs, McGraw Hills
3. Computer Architecture & Organization– John P Hayes, Mc Graw Hill.
4. Structured Computer Organization – Andrew S. Tanenbaum, 4th Edition, Pearson/PHI.

Paper 6 core: IC3CRP03 Software Lab 3

Programming in Python- Lab

1. Programs using conditional Branching and looping
2. Programs using functions and strings
3. Programs using Lists, Dictionaries, tuples and sets.
4. Program for traversing dictionaries
5. Programs using class, Inheritance and Exceptions.
6. Programs using files.
7. Arrays and Matrix using NumPy.
8. Plotting and Visualization using Matplotlib (Line, bar chart, pie chart etc..).
9. Data manipulations (data series and data frames) using Pandas
10. Simple programs using GUI

LAB Questions Using R language

LIST OF EXPERIMENTS

1. R PROGRAM TO ADD TWO VECTORS.

The program covers:

a. R Vector

b. R Operators

2 FIND SUM, MEAN AND PRODUCT OF VECTOR IN R USING BUILT-IN FUNCTIONS

The program covers:

a. R Variables and Constants

b. R Functions

3 R PROGRAM TO PRINT THE FIBONACCI SEQUENCE BY TAKING INPUT FROM THE USER

The program covers:

a. R if…else Statement

b. R while loop

4 R PROGRAM TO FIND THE FACTORS OF A NUMBER

The program covers:

a. R Functions

b. R for Loop

5 CALCULATOR APPLICATION in R

The program covers:

a. Using with and without R objects on console

b. Using mathematical functions on console

6 READING AND WRITING DIFFERENT TYPES OF DATASETS

a. Reading different types of data sets (.txt, .csv) from web and disk and writing in file in specific disk location.

b. reading Excel data sheet in R.

c. reading XML dataset in R

7 Solve this system using R:

Compute the inverse of the resultant matrix

Compute the determinant matrix.

Compute the Eigenvalues/eigenvector.

$x1 + x2 = 2$

$-x1 + x2 = 4$

8 Solve the system of linear equations using R.

$5x + y = 15$

$10x + 3y = 9$

9 Write an R program to access the element at

3rd column and 2nd row, only the

3rd row and only the 4th column of a given matrix.

10 DESCRIPTIVE STATISTICS IN R

a. Write an R script to find basic descriptive statistics using summary, str, quartile function on mtcars& cars datasets.

b. Write an R script to find subset of dataset by using subset (), aggregate () functions on iris dataset.

Semester IV

Paper 1: English II –As Approved by BoS of English

Paper 2: Complementary Course in Electronics

MICROPROCESSORS

(*Submitted as suggested by the BOS of Electronics (PG and UG) of Mahatma Gandhi University*)

Course Outcome

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

CO1: Assess and solve basic binary math operations using the microprocessor and explain the 8086 microprocessors internal architecture and its operation.

CO2: Apply knowledge and demonstrate programming proficiency using the various addressing modes and data transfer instructions of the microprocessor 8086.

CO3: Discuss operations of 8087, 8089, 8255 etc.

CO4: Get an idea of advanced processors.

(*PSO that can be linked to: Understand the core concept of computer science for applying and developing logic circuits, understanding processors and assembly language programing*)

Module I: Introduction to Microprocessor **5 hours**

Overview of binary, hexadecimal and BCD arithmetic, Introduction to microprocessors, Microprocessor Evolution, Overview of microcomputer structure and operation. Introduction to computer programming languages and Assembly language programing**.** Introduction to16-bit Processors – 8086/8088. *refer text book 1 and 3*

Module II: 8086 Architecture **14 hours**

Pins and signal descriptions of 8086 processor, Introduction to Maximum and Minimum mode operation, 8086 Architecture, Register organization of 8086, Physical Memory organization,

Addressing Modes. 8086, General Bus operation, Read machine cycle, Write machine cycle, Memory and I/O Mapping, 8086 interrupts, interrupt Types, Introduction to Stack, Stack Structure of 8086, Timer/counter. *refer text book 1 and 3*

Module III: Instruction set and programming of 8086 16 hours

Machine language instruction formats , 8086 instruction set, Assembler directives, Machine level programming/coding steps, Use of MASM (Microsoft Macro Assembler)/ TASM

(Turbo Assembler), Assembly language programming examples (arithmetic, logical, stack and delay subroutines). Software Development with Interrupts, Introduction to Subroutines, Recursion, MACROS. *refer text book 1*

Module IV: Peripherals and interfacing 13 hours

Numeric Processer 8087-Architecture and registers, I/O Processor 8089-Architecture, Memory Interfacing, RAM interfacing, Interfacing I/O ports, Programmable peripheral

Interface 8255, Modes of operation of 8255 *refer text book 1*

Module V: Advanced Architecture 6 hours

80286, 80386, 80586(Pentium) processors- System Architecture and salient features (Detailed study not needed). Introduction to Single Board Computers- Arduino, Raspberry Pi- Features *refer text book 1 and 3*

Text Books:

1. Advanced microprocessor and Peripherals - A.K.Ray and K.M.Bhurchandi, TMH, 2000
2. Micro Processors & Interfacing – Douglas V. Hall, 2007.
3. O'Reilly Raspberry Pi Cookbook- Simon Monk, O'Reilly.

Reference Books:

1. The Intel Microprocessor, Architecture, Programming and Interfacing - Barry B. Brey, 6e, Pearson Education / PHI, 2003.
2. Microcomputer systems-The 8086 / 8088 Family – Y.C. Liu and A. Gibson, 2nd edition, PHI -2003.
3. The 8086 Microprocessor: Programming & Interfacing the PC – Kenneth J Ayala, CENGAGE Learning, 2011.

Paper 3 Core: IC4CRT11 Data Mining

Instructional Hours/Week: 4
Total Instructional Hours: 72
Credits: **4**

Aim: To introduce the basic concepts and techniques of Data mining

Course Objectives:

To identify the scope and essentiality of Mining
To analyse data, choose relevant models and algorithms for respective applications.
To develop research interest towards advances in data mining

Module I

Introduction: What is Data mining? Data Mining Tasks, KDD process, Major issues in Data Mining, Data objects and Attribute types- Nominal, Binary, Ordinal and Numeric attributes, Measuring the central tendency- Mean, Median and Mode. Data Warehouse.

Data Pre-processing: Needs of Pre-processing the Data, Data Cleaning- Missing Values, Noisy Data, Data Cleaning as a Process. Data Integration- Redundancy and correlation analysis, Data Reduction- Attribute Subset Selection, Dimensionality Reduction, Numerosity Reduction, PCA. Data Transformation strategies, Data transformation by Normalization, Discretization by Binning, Histogram Analysis (16 hrs)

Module II

Association Analysis- Frequent patterns, Basic terminology in association analysis- Binary representation, Itemset and support count, Association Rule, Support and Confidence, Frequent Item set generation- The Apriori Algorithm, Generating Association Rules from Frequent Itemsets, FP Growth algorithm, From Association Analysis to Correlation Analysis.(14 hrs)

Module III

Classification- Basic concepts, General approach to classification, Nearest neighbor models, Cross validation and re-sampling methods- kfold cross validation, Boot strapping, Measuring classifier performance- Precision, recall, ROC curves. Bayes Theorem, Bayesian classifier, Decision Trees- Entropy, Information Gain, Tree construction, ID3, Issues in Decision Tree learning- Avoiding Over-fitting, Reduced Error Pruning, The problem of Missing Attributes, Gain Ratio, Classification by Regression (CART) (12 hrs)

Module IV

Neural Networks- The Perceptron, Activation Functions, Training Feed Forward Network by Back Propagation.

Support Vector Machine- Optimal Separating hyper plane, Soft-margin hyper plane, Kernel trick, Kernel functions. Ensemble Learning, Combining multiple learners, Ways to achieve diversity, Model combination schemes, Voting, Bagging, Boosting. (14 hrs)

Module V

Cluster Analysis: Introduction, Basic Clustering methods- Partitioning methods- k-Means and k-Medoid. Hierarchical Methods - Agglomerative and Divisive Hierarchical Clustering. Density Based Methods - DBSCAN, OPTICS. Grid Based- STING, CLIQUE, Outlier Analysis- what are outliers, Types of outliers, Outlier detection methods - Statistical Distribution-based Outlier Detection, Distance-Based Outlier Detection. Mining other kinds of data – Mining spatial data, mining multimedia data, mining text data, mining web data. (16 hrs)

Text Book

1. Jiawei Han & Micheline Kamber , Data Mining, Concepts and Techniques, , 3rd Edition.
2. Pang Ning Tan, Michael Steinbach and Vipin Kumar, Introduction to Data Mining, Pearson India Education Services.

Reference Text

1. Arun K Pujari, Data Mining Techniques, , University Press
2 Sam Anahory & Dennis Murray, Data Warehousing in the Real World, Pearson Education, Asia.
3. Paulraj Ponnaiah, Data Warehousing Fundamentals,Wiley Student Edition

Paper 4 Core : IC4CRT12 Software Engineering

Instructional Hours/Week: 4
Total Instructional Hours: 72
Credits: **3**

Aim: To provide better understanding about the basic concepts of Software Engineering

Objective:

Upon the completion of the course, students should be able to

- Recognize the importance of basic processes in software Development life cycle.
- Understand the various activities associated with different models and their significance.

- Familiarize the requirements in engineering and systematic approach in classical software design and development techniques.
- Familiarize with various software testing techniques and tools.
- Perceive the importance of Software Maintenance

Unit I: **Introduction**: Evolution, Types of software development products; **Software life cycle models**: A few basic concepts, Waterfall model and its extension, Agile development models, Spiral model, Comparison of different life cycle models (14 hrs)

Unit II: **Software Project Management**, Project Planning, Metrics for project size estimations, Project Estimation Techniques, Basic COCOMO model, Scheduling, Organization structure, Team structure , Staffing, Risk Management, **Software Requirements Analysis and Specification**: Requirements gathering and analysis, Software Requirements Specification(SRS) (14 hrs)

Unit III: **Software Design**: overview of the design process, How to characterize a good software design, Cohesion and Coupling, Layered arrangements of modules, Approaches to software design, **Function oriented design**: Overview of SA/SD Methodology, Structured analysis, Developing the DFD model of a system, Structured Design, **User Interface design**: Characteristics of a good user interface, Basic concepts, Types of user interfaces (14 hrs)

Unit IV: **Coding and Testing**: Coding, Code review, Software documentation, Testing, Unit testing, Black box testing, white box testing: Basic concepts, Debugging Integration testing, system testing, **Software Reliability and quality management**: Software reliability, Software quality (14 hrs)

Unit V: **Software maintenance**: Characteristics of software maintenance, Software reverse engineering**,** Software process models, Estimation of maintenance cost**, Software Reuse:** Basic issues in any Reuse Program, A Reuse approach, Reuse at Organization level , **Emerging Trends**: Client Server Software, Client Server architectures, CORBA, Service Oriented Architecture (SOA), Software as a Service (SaaS). (16 hrs)

Book of Study:

Fundamentals of Software Engineering, Fifth Edition by Rajib Mall, PHI Learning Pvt. Ltd., February 2019.

References:

1. Software Engineering 10th Edition by Ian Sommerville, PEARSON INDIA, October 2018.

2. Software Engineering – a Practitioner's approach Seventh Edition by Roger S Presman, 7th edition, McGraw Hill. 2017.

Paper 4 Core IC4CRT13: Basics of Artificial Intelligence

Instructional Hours/Week: 4
Total Instructional Hours: 72
Credits: **3**

Upon successful completion of this course students should be able to:

Explain the basics of AI.

Identify appropriate AI methods to solve a given problem.

Will be able understand the concept of differentiation and its applications

Should have sound knowledge in the field of probability and correlation and regression

U**nit I**: Introduction to AI

Introduction to AI-Problem formulation, Problem Definition -Production systems, Control strategies, Search strategies. Problem characteristics, Production system characteristics -Specialized productions system- Problem solving methods – Problem graphs, Matching, Indexing and Heuristic functions -Hill Climbing-Depth first and Breath first, Constraints satisfaction – Related algorithms, Measure of performance and analysis of search algorithms. (14 hrs)

Unit II: Representation of Knowledge

Game playing – Knowledge representation, Knowledge representation using Predicate logic, Introduction to predicate calculus, Resolution, Use of predicate calculus, Knowledge representation using other logic-Structured representation of knowledge. (14 hrs)

Unit III:

Differentiation, Limits and continuity rules of differentiation, Derivatives, Scalar derivatives, Partial derivatives, Differentiation of univariate functions, Partial differentiation and gradients, Gradient of vector valued function. Gradient of matrices. Optimization using gradient functions, constrained optimization, and Lagrange multipliers. Convex optimization. Back propagation in neural networks, implementation and application. (15 hrs)

Unit 1V:

Generating Functions of Sequences, Calculating Coefficients of Generating Functions, Recurrence Relations, Solving Recurrence Relations by Substitution and Generating Functions, The Method of Characteristic Roots, Solutions of Inhomogeneous Recurrence Relations, Complexity calculations of prominent algorithms.(15 hrs)

Unit V:

Probability, basics, Conditional Probability, Bayes Theorem, Distributions - Binomial, Poisson, normal distributions, and related problems. Descriptive Statistics, Regression, and correlation, Bayesian classification, implementation, applications. (14 hrs)

Text Book

1. Artificial Intelligence, Deepak Khemani, Tata Mc Graw Hill Education.
2. Kenneth H. Rosen, "Discrete Mathematics And Its Applications", 7th Ed,McGrawHill, 2012.
3. Erwin Kreyszig, "Advanced Engineering Mathematics", Wiley India, 9th Edition 2011.
4. Walpole, R. E., Myers, R. H., Myers S L & Keying Ye, 'Probability and Statistics for Engineers and Scientists'. 8th ed, Pearson Education, 2007.

Paper 5: IC4CEP01 Complementary Lab

(*As suggested by the BOS of Electronics (PG and UG) of Mahatma Gandhi University*)

Course Outcome

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

CO1: Analyze assembly language programs using appropriate assembler.

CO2: Construct a maintainable assembly language program for an algorithm.

ASSEMBLY LANGUAGE PROGRAMING (8086 based)

(SYLLABUS)

1. Simple Arithmetic Calculations
2. Conditional statements
3. Control statements
4. Loop and arrays

5. Character strings
6. Subroutines and Stack Operations

(*Minimum two programs from each section, total 20 programs to be carried outcompulsorily to appear for external examination)*

Sample List of Programs

Write Assembly Language Program using 8086

1. Data transfer using direct and indirect addressing
2. Block data transfer from one section of memory to another section of memory
3. Addition and subtraction of two unsigned numbers.
4. 16 bit addition.
5. Multiplication of two numbers.
6. Multiplication by shift rotate and add method.
7. Division of two numbers.
8. Checking specific bits in a number.
9. Finding the number of negative numbers in a dataset.
10. Finding largest number in a dataset.
11. Finding smallest number in a dataset
12. Sorting in ascending order.
13. Sorting in descending order.
14. BCD addition and subtraction.
15. BCD to HEX conversion
16. Finding the square of a given number.
17. Checking parity of a given number.
18. Square of a number
19. Find the Square from the hexadecimal look uptable
20. Hexa decimal to ASCII conversion
21. ASCII to Hexa decimal conversion
22. Occurrence of a number in a dataset
23. Binay to BCD conversion
24. BCD to binary conversion
25. Factorial of a number
26. Moving a string from one location to another
27. Checking of password
28. Check for palindrome

Paper 6 Core: IC4CRP04 Software Lab IV.

Software Lab IV- Data Mining Implementation using Python & R

1. Implement Apriori algorithm for frequent item set generation using Python.
2. Implementation of Classification in Python using
    a. KNN
    b. Decision Tree
    c. SVM
    d. Neural Networks
3. Implementation of Clustering in Python using
    a. K-means
    b. K-medoid
4. Regression Model using R
    a. Import a data from web page
    b. Name the dataset and now do Logistic Regression to find out relation between variables that are affecting the admission of a student in an institution based on his/her GRE score, GPA obtained and rank of the student.
    c. Also check whether the model if fit or not.
5. Classification model using R using above dataset
    a. Install relevant package for classification
    b. Choose classifier for classification problem
    c. Evaluate the performance of classifier
6. Implementation of Correlation and Covariance analysis using R
    a. Find the correlation matrix
    b. Plot the correlation plot on dataset and visualize an overview of relationships among data on iris dataset.
    c. Analysis of covariance: Variance (ANOVA), if data have categorical variables on iris data.

Semester V

Paper 1 Core: IC5CRT14 Principles of Machine Learning

Instructional Hours/Week: 3
Total Instructional Hours: 54
Credits: **4**

Unit I:

Introduction to Machine Learning – Machine learning basics, Types of machine learning, Applications of Machine Learning, Basic types of data in Machine learning, Data pre-processing, Predictive Vs Descriptive models, training a model, training versus testing, cross validation, overfitting & underfitting, Bias variance tradeoff, error measures, evaluating performance of a model. Introduction to feature engineering, ML tools in Python (12 hrs)

Unit II:

Linear Regression – Problem formulation, Parameter Estimation, Bayesian Linear Regression, Multiple linear regression, fitting simple linear and multiple linear regression equations with examples, regularization techniques, case study and implementation. (10 hrs)

Unit III:

Logistic Regression - Interpreting Parameters in Logistic Regression, Inference for Logistic Regression, Logistic Models with Categorical Predictors, Multiple Logistic Regression, Fitting Logistic Regression Models and its implementation using real life examples.(10 hrs)

Unit IV:

Normal distribution, Standard normal distribution, Sampling distributions, Hypothesis testing – Large sample mean test, Chi square test, t-test, Analysis of variance. (10 hrs)

Unit V:

Introduction to Categorical variables, Independent and dependent variables. Distributions for Categorical Data, Statistical Inference for Categorical Data, Statistical Inference for Binomial Parameters, Statistical Inference for Multinomial Parameters, Bayesian Inference for Binomial and Multinomial Parameters, Maximum likelihood function and maximum likelihood estimates, loss functions in neural networks, implementation, real life applications. (12 hrs)

TEXT BOOKS

1. Machine Learning, Saikat Dutt, Subramanian Chandramouli, Amit Kumar Das, Person, 2020.
2. Walpole, R. E., Myers, R. H., Myers S L & Keying Ye, 'Probability and Statistics for Engineers and Scientists'. 8th ed, Pearson Education, 2007.
3. Peter Bruce, "Practical Statistics for Data Scientists, 2015
4. Marc Peter Deisenroth, A. Aldo Faisal, Cheng Soon Ong, "Mathematics for Machine Learning", 2020.
5. Aurelien Geron, "Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow" Powered by Jupyter, published by O'Reilly Media

Paper 2 Core IC5CRT15: Web Application Development Using PHP

Instructional Hours/Week: 4
Total Instructional Hours: 72
Credits: **3**

Aim: To develop programming kills of students in Full Stack Development.

Course objective: Upon successful completion of this course, students should be able to:

- Develop web applications using php and MySQL database.
- Use java scripts and jQuery in client side
- Use CSS concepts in Webpage designing

Unit -1

Introduction to PHP- Structure of PHP-Comments, Basic Syntax, Variables, Variable Assignment, Variable Typing, Constants, Predefined Constants, echo vs print Command, Functions, Variable Scope.

Expressions and Control Flow in PHP- Expressions, Conditionals- if Statement, else Statement, elseif Statement, switch Statement? Operator, Looping- while Loops, do...while Loops, for Loops, break, continue Statement.

PHP Functions and Objects- PHP Functions- Defining a Function, Returning a Value, Returning an Array, Passing Arguments by Reference, Returning Global Variables, Including and Requiring Files, PHP Objects- Declaring a Class, Creating an Object, Accessing Objects, Cloning Objects, Constructors, Destructors, Writing Methods, Declaring Properties, Declaring Constants, Property and Method Scope, Static Methods, Static Properties, Inheritance (16 hrs)

Unit -2

PHP Arrays -Basic Access- Numerically Indexed Arrays, Associative Arrays, Assignment Using the array Keyword, foreach Loop, Multidimensional Arrays, Array Functions .Accessing MySQL Using PHP -Connecting to a MySQL Database, $_POST Array, create, insert, delete, update, select operations in MySQL database using PHP, Form Handling - Building Forms - Retrieving Submitted Data, Default Values, Input Types, HTML5 Enhancements- autocomplete Attribute, autofocus Attribute, placeholder Attribute, required Attribute, Override Attributes, width and height Attributes, min and max Attributes, step Attribute, form Attribute, list Attribute, color Input Type, number and range Input Types, Date and Time Pickers (14 hrs)

Unit -3

Exploring JavaScript - JavaScript and HTML Text- Using Scripts Within a Document Head, Including JavaScript Files, Using Comments, Semicolons, Variables- String Variables, Numeric Variables, Arrays, Variable Typing, Functions, Global Variables, Local Variables, Document Object Model document.write, console.log, alert, Writing into Elements, with Statement, onerror, Using try...catch, Conditionals, Loops.

JavaScript Functions, Objects, and Arrays - JavaScript Functions- Defining a Function, Returning a Value, Returning an Array, JavaScript Objects- Declaring a Class, Creating an Object, Accessing Objects, prototype Keyword, JavaScript Arrays- Numeric Arrays, Associative Arrays, Multidimensional Arrays, Array Methods, Asynchronous Communication - XMLHttpRequest, Sending XML Requests (14 hrs)

Unit- 4

Introduction to CSS - Importing a Stylesheet, Importing CSS from Within, Embedded Style Settings, Using IDs, Using Classes, Using Semicolons, CSS Rules- Multiple Assignments, Using Comments, Style Types- Default Styles, User Styles, External Stylesheets, Internal Styles, Inline Styles, CSS Selectors- Type Selector, Descendant Selector, Child Selector, ID Selector, Class Selector, Attribute Selector, Universal Selector, Selecting by Group, CSS Cascade-Stylesheet Creators, Stylesheet Methods, Stylesheet Selectors, Measurements, Fonts and Typography- font-family, font-style, font-size, font-weight, Managing Text Styles- Decoration, Spacing, Alignment, Transformation, Indenting, CSS Colors- Short Color Strings, Gradients, Positioning Elements- Absolute Positioning, Relative Positioning, Fixed Positioning, Pseudoclasses, Box Model and Layout- Setting Margins, Applying Borders, Adjusting Padding, Object Contents. (14 hrs)

Unit -5

Introduction to jQuery- Including jQuery, jQuery Syntax, Avoiding Library Conflicts, Selectors, *css* Method, Element Selector, ID Selector, Class Selector, Combining Selectors, Handling Events, Event Functions and Properties- blur and focus Events, click and dblclick Events, keypress

Event, mousemove Event, Other Mouse Events, Alternative Mouse Methods, submit Event, Special Effects, Hiding and Showing, toggle Method, Fading In and Out, Sliding Elements Up and Down, Manipulating the DOM- text vs html Methods, val and attr Methods, Adding and Removing Elements, Dynamically Applying Classes, Modifying Dimensions, DOM Traversal- Parent Elements, Child Elements, Sibling Elements, Selecting the Next and Previous Elements, Traversing jQuery Selections, *is* Method, Using jQuery Without Selectors- $.each Method, $.map Method, Asynchronous Communication . (14 hrs)

Text Book:

1. Learning PHP, MySQL & JavaScript, Robin Nixon 5th Edition, O'Reilly

References:

1. Learn PHP 7: Object-Oriented Modular Programming using HTML5, CSS3, JavaScript, XML, JSON, and MySQL- Steve Prettyman , Apress
2. PHP, MySQL, JavaScript & HTML5 All-in-One For Dummies - Steve Suehring and Janet Valade, Wiley
3. Beginning JavaScript with DOM Scripting and Ajax From Novice to Professional , Christian Heilmann, Apress
4. Beginning jQuery: From the Basics of jQuery to Writing your Own Plug-ins- Jack Franklin Russ Ferguson, Second Edition, Apress
5. Sams Teach Yourself HTML, CSS & JavaScript Web Publishing in One Hour a Day, Seventh Edition

Paper 3 Core: IC5CRT16 Programming in Java

Instructional Hours/Week: 4
Total Instructional Hours: 72
Credits: **3**

Aim:

This course explains the fundamentals of Java programming language which leads to the understanding of object oriented programming concepts. After completion of this course students will be able to solve real world problems by the application of concepts and principles of Java programming language.

Objectives:

To familiarize input/output and file handling.
To inculcate concepts of GUI programming using Swing
To demonstrate exception handling, use of packages and multithreading
To introduce database connectivity and network based Java applications

Unit I : Object Oriented Programming Concepts and Basics of Java.

Java Programming Environment – JDK, Java Virtual Machine, Bytecode, Features of Java Arrays –One Dimensional Array, Multi-dimensional Array , Object Oriented Programming Concepts- (Objects and Classes, Encapsulation, Inheritance, Polymorphism), Method Overloading, Method Overriding, Dynamic Method Despatch , Constructors- Constructor Overloading , this, super, final, abstract and static Keywords (12 hrs)

Unit II : Input/Output and File Handling

Interfaces- Defining an Interface, Implementing Interface, Extending Interfaces. String - String Handling Fundamentals, Comparison of String and StringBuffer Class, Special String Operations- Character Extraction, String Comparison, Searching String, Modifying a String, String Copy ,Input and Output Streams – Byte Stream , Character Stream. File Management - Reading and Writing Files (FileInputStream and FileOutputStream Classes), (14 hrs)

Unit III : Packages; Exception Handling and Thread

Packages – Defining Packages, Built in Packages(java.lang, java.util, java.io, java.net, javax.swing), Importing Packages, Implementation of User Defined Packages, Access Protection in Java, Exception Handling - try, catch, throw, throws and finally Statements, Java's Built-in Exceptions, Creating User Defined Exceptions. Threads- Thread Lifecycle, Thread Priorities, The Thread Class, Runnable Interface, Creating a Thread – Implementing Runnable, Extending Thread, Inter Thread Communication, Suspending Resuming and Stopping Threads.(16 hrs)

Unit IV: GUI Programming

Basic Event Handling – Delegation Event Model, Important Event Classes And Listener Interfaces, Handling Mouse and Keyboard Events, Adapter Classes, Swing -Window Fundamentals – Class Hierarchy, Frame, Creating a Simple Window Based Application, ImageIcon, JLabel, JTextField, JTextArea, JButton, JCheckBox, JRadioButton, JList, JComboBox, JTable, JTabbedPane, JScrollPane, Layout Management – The FlowLayout, BorderLayout, GridLayout, CardLayout (16 hrs)

Unit V: Database and Networking

Networking Basics- Networking Classes and Interfaces, InetAddress, TCP/IP Client Sockets, URL Connection, TCP/IP ServerSockets, JDBC - The Design of JDBC, JDBC Configuration, Executing SQL Statements- Scrollable and Updatable ResultSets, RowSets, Transactions. (14 hrs)

Text book :Herbert Schildt ,Java 2 The Complete Reference, Tata McGraw Hill (5^{th} Edn.)

Reference Text

1. James. P. Cohoon, Programming java 5.0, , Jack. W. Davison (Tata McGraw Hill)
2. C Thomas Wu, An introduction to Object Oriented Programming with Java, , Tata McGraw Hill, (2006)

3. Wigglesworth and McMillan ,Java Programming: Advanced Topics, , Cengage Learning India, 3rd Edn.
4. Bernard Van Haecke, JDBC:Java Database Connectivity, , IDG Books India (2000)

Paper 4: Core: IC5CRT17: IT and Environment

Instructional Hours/Week: 4
Total Instructional Hours: 72
Credits: **4**

Unit 1: Introduction to Internet and Environment:

Internet- Internet as a knowledge repository, academic search techniques, creating cyber presence. Academic websites. Multidisciplinary nature of environmental studies -Definition, scope and importance, Need for public awareness. (10 hrs)

Unit 2: Impact of IT in E-Learning:

Introduction to use of IT in teaching and learning, Learning Management SystemMoodle, Edmodo, etc. Academic services– A note on INFLIBNET, NPTEL, NICNET. (12 hrs.)

Unit 3: IT and Society:

IT & Society- issues and concerns- digital divide, IT & development, the free software movement. IT industry: new opportunities and new threats, software piracy, cyber ethics, cybercrime, cyber threats, cyber security, privacy issues, cyber laws, cyber addictions, information overload, health issues guidelines for proper usage of computers, internet and mobile phones. Impact of IT on language & culture. (18 hrs.)

Unit 4: E-waste and Green Computing:

E-waste- Problems- Solutions-Impact of e-waste in living beings and environment- a study on e-waste management in India. Green computing, definition, meaning, scope. Green computing in India. (18 hrs.)

Unit 5: Human Rights:

An Introduction to Human Rights, Meaning, concept and development –History of Human Rights- Different Generations of Human Rights- Universality of Human Rights- Basic International Human Rights Documents - UDHR ,ICCPR,ICESCR.-Value dimensions of Human Rights

Human Rights and United Nations : Human Rights co-ordination within UN system- Role of UN secretariat The Economic and Social Council- The Commission Human Rights-The Security Council and Human rights The Committee on the Elimination of Racial Discrimination- The Committee on the Elimination of Discrimination Against Women- the Committee on Economic, Social and Cultural Rights- The Human Rights Committee- Critical Appraisal of UN Human Rights Regime.

Human Rights National Perspective : Human Rights in Indian Constitution – Fundamental Rights- The Constitutional Context of Human Rights-directive Principles of State Policy and Human Rights- Human Rights of Women- children –minorities- Prisoners- Science Technology and Human Rights- National Human Rights Commission- State Human Rights Commission- Human Rights Awareness in Education. (18 hrs.)

Case Study:

The students need to view the film "**Samaksham**", a film on environment produced by Mahatma Gandhi University Creations and submit a compulsory assignment reviewing film. The review is considered for internal mark assessment.

References :

1. K.L. James, The Internet: A User's Guide 2nd Revised edition, PHI publication.

2. Bharucha Erach, Text Book of Environmental Studies for undergraduate Courses. University Press, IInd Edition 2013 (TB)(https://www.ugc.ac.in/oldpdf/modelcurriculum/env.pdf).

3. Barkha and U Rama Mohan, Cyber Law & Crimes, 3rd Edition, Asia Law House.

4. Rakesh Johri, E-waste: Implications, regulations, and management in India and current global best practices, Teri publications.

5. Alan Evans, Kendall Martin, Mary Anne Poatsy, Technology in Action, Pearson.

Project – General guidelines for Phase I and II

Each student needs to undertake a project work to implement various phases of Software Development. The project work is divided into two phases, Phase I in 5th and Phase II in the 6th Semester.

The Phase I includes Problem identification and statement, system study and system design. For this he/she needs to identify a problem which is not yet automated. Analyze the Manual System existing there and suggest a framework of software befitting the problem. In the fifth semester (Phase I), system must be developed showing the data flow and needed DFD's and Database design. Evaluation of Project-Phase I will be internal for which the student needs to submit a spiral bound report of their work and appear before a team of faculty members formed by the head of the Department. The team should comprise of the Project Guide and two other Faculty members from the Department.

The Phase II of the project work in the Sixth semester is purely meant for development of software implementing the project identified and designed in Phase I. Student needs to develop a software using any of the Language or Package they have studied in their syllabus. Usage of any other language/package needs to get approved by the Committee/ Project Guide/HoD of the Department No student should be allowed to change the project work in Phase II and they need to develop the software for the work they identified in phase1. The internal mark for the Project work is to be awarded based on the student's performance in Phase I and II (50 percent Weightage for both the semesters). The Sixth semester project evaluation will be of External Nature along with its internal component. Students have to submit a hard bounded report for the evaluation.

Paper 5 Core: IC5CRP05 Software Lab V

Lab I Advanced Java Programming

1. Basic Concepts and File Handling
   - 1.1. Inheritance, Polymorphism
   - 1.2. Constructors
   - 1.3. Interface
   - 1.4. Package
   - 1.5. One Dimensional and Two Dimensional Array Manipulation
   - 1.6. String Handling (Character Extraction, String Comparison, Searching String, Modifying a String, String Copy)
   - 1.7. Exception (Built-in and User Defined)
   - 1.8. Thread (Using Runnable Interface and Thread Class)
   - 1.9. File management (File reading, Writing, Appending and Content Replacing)

Web Application Development Using PHP: Lab

Develop programs for implementing the following concepts

1. Expressions and Control Flow in PHP.
2. PHP Functions -Returning a Value, Returning an Array, Passing Arguments by Reference, Returning Global Variables
3. Constructors, Destructors, Inheritance
4. PHP Arrays - Numerically Indexed Arrays, Associative Arrays, foreach Loop, Multidimensional Arrays.
5. MySQL Database - create, insert, delete, update, select operations using HTML form
6. HTML5 Enhancements- autocomplete Attribute, autofocus Attribute, placeholder Attribute, required Attribute, Override Attributes, width and height Attributes, min and max Attributes, step Attribute, form Attribute, list Attribute, colour Input Type, number and range Input Types, Date and Time Pickers
7. JavaScript - Conditionals, Loops, Functions, Objects, and Arrays
8. Asynchronous Communication - XMLHttpRequest, Sending XML Requests,
9. CSS Selectors- Type Selector, Descendant Selector, Child Selector, ID Selector, Class Selector, Attribute Selector, Universal Selector, Selecting by Group.
10. CSS- Fonts and Typography- font-family, font-style, font-size, font-weight, Managing Text Styles- Decoration, Spacing, Alignment, Transformation, Indenting.
11. CSS Colors- Short Color Strings, Gradients
12. Positioning Elements- Absolute Positioning, Relative Positioning, Fixed Positioning.
13. Pseudoclasses,
14. Setting Margins, Applying Borders, Adjusting Padding.
15. JQuery- Selectors, *css* Method, Element Selector, ID Selector, Class Selector, Combining Selectors
16. JQuery event handling
17. Special Effects, Hiding and Showing, toggle Method, Fading In and Out, Sliding Elements Up and Down.
18. Manipulating the DOM
19. Using jQuery Without Selectors- $.each Method, $.map Method
20. Asynchronous Communication.

Semester VI

Paper1 Core: IC6CRT18 Linux and Shell Programming

Instructional Hours/Week: 3
Total Instructional Hours: 54
Credits: **3**

Aim: To introduce the concept of Open source Software and basic features of Linux and Shell Programming

Course objective: Upon completion of the course, students will be able to gain working knowledge in Linux environment, a clear view on Linux file system, process scheduling in Linux, facilities for user creation and management and basics of shell programming.

Unit 1: Open Source Software : Free software foundation, Freedoms in free software, GNU project, Introduction and development of Linux, , advantages of Linux, Hardware requirement, Installing Linux, Linux File System overview, Linux Architecture, Boot Process, Kernel, shell - the user interface, GUI and CLI commands, Usage of input-output redirection (>, >> etc.), Basic Commands in Linux – creating directories, changing directories, listing directory contents, file related commands (create and edit text files, renaming, copying, deleting), Introduction to Shells, Different shells and their features, Online manuals in Linux, the Man command. (10 hrs)

Unit 2: The Linux File system: Partitioning the disk, fDisk command, Important files and directories in Linux, the hierarchical file system. The root directory / and its important sub directories /bin, /etc, /dev /lib, /boot /home. /mnt, /tmp, /user. Navigating through the file system. Absolute and relative pathnames. The disk related commands, df, du, creating new partitions in Linux, deleting a partition, mounting and unmounting file systems , file types, file related commands Find, touch, cat etc. searching for a pattern, concept of wild cards and regular expressions , grep, egrep commands. (10 hrs)

Unit 3: Process Management in Linux The Concept and properties of Processes, Creating processes, the Parent processes and child processes, PID's and its relevance, Killing processes and sending signals to a process (kill, killall, xkill), How to start and monitor processes, Identify CPU/memory intensive processes, adjust process priority, Processor scheduling in Linux. The Batch command, The at Command, nohup command, File processing commands, wc, cut, paste, sort , Mathematical Commands expr and factor. Different editors in Linux, Consol based editors and GUI based editors and comparison of basic features (Vi, ed, emacs, gEdit etc..) .(12 hrs)

Unit 4: Users and Group Management, useradd, usermod, userdel, groupadd, groupmod, groupdel. Adding a New User Account, User Private Groups, Modifying / Deleting User Accounts, Group Administration, Password Aging Policies, Switching Accounts, passwd command, logging in as Super user, Networked Users and communication, Authentication, Configuration, Default File Permissions, Changing file ownership (chown), Changing file group ownership (chgrp), Permissions on files, Permissions on directories, How permissions are applied, Changing permissions (chmod),Access Control Lists (ACLs). (10 hrs)

Unit 5: Shell Programming: The role of shells in the Linux environment, : The bash shell, Shell commands, Other standard shells, Write a simple shell script to welcome users, Comments in a script ,Setting up permissions on a script, debug and Execute a script, Variables in shell, The export statement, Unset shell and environment variables, Getting User Input Via Keyboard , Bash variable existence check, Customize the bash shell environments: Recalling command history, Path name expansion, Create and use aliases, The tilde expansion, Startup scripts-(Using aliases, Changing bash prompt, Setting shell options , Setting system wide shell options), Commonly Used Commands and Utilities (ls,rm,cat etc),Developing shell scripts for adding a User, Changing Password of users etc.. (12 hrs)

Text Books:

1. A Practical Guide to Linux Commands, Editors, and Shell Programming, 4th Edition**,** by Mark G. Sobell, Matthew Helme, Prentice Hall, 2018. ISBN: 978-0-13-477460-2.
2. The redhat Linux Bible: Christopher Negus: Wiley Dreamtech India
3. Unix Shell Programming: Yeshwant kanetkar. BPB publications.

Paper 2 Core: IC6CRT19 Neural Networks and Deep Learning

Instructional Hours/Week: 4
Total Instructional Hours: 72
Credits:4

Aim : Aim of this course is to familiarize students with the basic concepts and mathematical foundation of Neural Networks and Deep Learning

Outcome: The expected course outcomes are

1. To impart knowledge on Neural Network design.
2. To acquire advanced theoretical knowledge about the simulation of various machine learning models with Neural Network.
3. To develop skills in analyzing the Training Methods for Neural Networks.

4. To construct and compare the application of three important neural network models -Radial basis function Network, Recurrent Neural Network and Convolutional Neural Network.
5. To evaluate neural network systems for solving real world problems.

Unit 1: An Introduction to Neural Networks - Introduction, The Basic Architecture of Neural Networks, Single Computational Layer : The Perceptron , Training a neural Network with Back propagation, The secrets to the Power of function Composition, Common Neural architectures

Unit 2 : Machine Learning with Shallow Neural Networks - Introduction , Neural Architecture for Binary Classification Models, Neural Architecture for Multiclass Models, Back propagated Saliency for feature selection, Matrix factorization with Auto encoders,

Unit 3: Training Deep Neural Networks – Introduction, Back propagation, set up and initialization issues, the vanishing and exploding Gradient problems, Gradient -Descent Strategies, Batch Normalization. Teaching Deep Learners to generalize – Introduction, The bias-variance trade-off, Generalization issues in model tuning and evaluation, Penalty-based regularization, ensemble methods, Early stopping, unsupervised pretraining, continuation and curriculum learning, parameter sharing,

Unit 4 : Radial basis function Networks – introduction, Training an RBF network, Variations and special case of RBF networks, Relationship with kernel methods, Recurrent Neural Networks- The architecture of recurrent Neural networks, The challenges of Training recurrent Networks, Echo-state networks, Long Short term memory

Unit 5: Convolutional Neural Networks- Historical Perspective and Biological Inspiration, Broader observation about convolutional neural networks, the basic structure of convolutional network, Training of convolutional network,

Text Books

1 Charu C Aggarwal - " Neural Networks and Deep Learning " – Springer -2018

Reference Books

1. Sandro Skansi, "Introduction to Deep Learning from Logical calculus to Artificial Intelligence", Springer , 2018.

2. Deep Learning, Ian Goodfellow and Yoshua Bengio and Aaron Courville, MIT Press, 2016

3. Artificial l Neural networks – B.Yegnanrayana -PHI publications – 2012

4. Simon Haykin – Neural Networks and Learning Machines - 2008

Paper 3: Core: IC6CRT20: Computer Networks

Instructional Hours/Week: 4
Total Instructional Hours: 72
Credits: **4**

Aim of the Course

Aim of the course is to provide a better understanding about the computer network and it also gives an insight into networking protocols

Expected course outcomes

Upon completion of this course, the students will be able to:

1. Understand the concepts of signals and OSI layer functions.
2. Discuss the process of Multiplexing, switching and difference between guided and unguided media in networks
3. Describe, analyze various data link layer protocols
4. Describe and analyze various network, and transport layer protocols
5. Have a basic knowledge of the use of cryptography and network security

Unit 1: Introduction to Networks, Data and signals-analog and digital, periodic analog signals, digital signals, bit rate, baud rate, bandwidth. Transmission impairments- attenuation, distortion and noise. Data communication protocols and standards, Network models - OSI model-layers and their functions. TCP/IP protocol suite. (14 hrs)

Unit 2: Bandwidth utilization Multiplexing: FDM, TDM, spread spectrum. Transmission Media-guided media and unguided media. Switching: message, Circuit and packet switched networks, datagram networks, virtual- circuit networks.(12 hrs)

Unit 3: Data link layer: Error Detection and Correction, Framing, flow and error control, Protocols - Noiseless channels (Simplest, Stop and Wait) and Noisy channels (Stop and Wait and Piggy Backing). Multiple Access Protocols. Random Access-ALOHA, CSMA. Wired LANs-IEEE standards, wireless LANs-Bluetooth, Cellular Telephony (16 hrs)

Unit 4: Network layer and Transport layer: Repeaters, Bridges, Gateways and routers. Logical addressing – IPV4 and IPV6 addressing, Internet protocol - IPV4 and IPV6. Connectionless and Connection Oriented Services: UDP and TCP. Congestion Control, Quality of Service. (16 hrs)

Unit 5: Application layer: HTTP, FTP, SMTP, DNS. Network security: Common Threats-Firewalls (advantages and disadvantages), Cryptography. (14 hrs)

Book of study:

1. B. A. Forouzan - Data communication and Networking, Fourth edition-,TMH

2. Andrew S Tanenbaum - Computer Networks ,Fourth Edition, Prentice Hall of India.

Paper 4 Core :IC6CRT21 Mobile Application Development Using Kotlin

Instructional Hours/Week: 3
Total Instructional Hours: 54
Credits: **3**

Aim: To introduce modern mobile application architecture and programming style

Course objective: Upon successful completion of this course, students should be able to:

- Kotlin programming
- Android programming using Kotlin

Unit -1

Kotlin Basics - Characteristics of Kotlin -Program Elements-Literals, Variables, Expressions and Statements, Keywords, Whitespace, Operators, Blocks, Comments, Basic Types-Numbers and Literal Constants, Characters, Booleans, Arrays, Strings and String Templates, Controlling Program Flow - ifs, when Statement, while Statement, for loops, Exception Handling, Handling Nulls. (10 hrs)

Unit -2

Functions- Declaring Functions, Single Expression Functions, Default Arguments, Named Parameters, Variable Number of Arguments, Extension Functions, Infix Functions, Operator Overloading.

Working with Types- Interfaces, Diamond Problem, Invoking Super Behaviour, Classes-Constructors, Inheritance, Properties, Data Classes, Visibility Modifiers, Access Modifiers, Object Declarations. (12 hrs)

Unit -3

Lambdas and Higher Order Functions- Higher Order Functions, Lambda and Anonymous Functions, Parameters in Lambda Expressions, Closures, *with* and *apply*.

Collections, Arrays - Arrays, Collections, Lists, Sets, Maps, Collections Traversal, Filter and Map.

Generics -Why Generics, Terminologies, Using Generics in Functions, Using Generics in Classes, Variance, Subclass vs Subtype, Reified Generics (12 hrs)

Unit -4

Activities and Layouts -Activity Class, Layout File, View and ViewGroup Objects , Containers, Event Handling-Introduction to Event Handling, Intents- Definition, Loose Coupling, Types of Intent- Explicit, Implicit Intents, Fragments,.

Running in the Background- Basic Concepts, UI Thread, Threads and Runnables , Handler Class, Async Task, Anko's doAsync . (10 hrs)

Unit -5

Data sharing and Storage -Shared Preferences, Sharing Data Between Activities, Internal Storage- Overview of File Storage, Internal and External Storage, Cache Directory.

Broadcast Receivers- Introduction to Broadcast Receivers, System Broadcast vs Custom Broadcast, Manifest Registration vs Context Registration, Basics of Broadcast Receivers , Implicit vs Explicit Broadcast Actions. (10 hrs)

Text Book:

1. Learn Android Studio 3 with Kotlin: Efficient Android App Development - Ted Hagos, Apress, Inc.

References:

2. Android Development with Kotlin - Marcin Moskala, Igor Wojda, Packt
3. Kotlin In-Depth [Vol-I] -A Comprehensive Guide to Modern Multi-Paradigm Language - Aleksei Sedunov , BPB Publications
4. Learn Kotlin for Android Development- The Next Generation Language for Modern Android Apps Programming , Peter Spath, Apress.

5. Programming Kotlin Applications- Building Mobile and Server-Side Applications with Kotlin, By Brett McLaughlin, Wiley

Paper 5: Core: IC6CPP02: Project Phase II (refer Phase1)

Paper 6: Core: IC6CRP06: Software Lab VI

Software Lab - Mobile Application Development using Kotlin

Develop GUI programs for android operating system and implement the following concepts

1. Lambdas and Higher Order Functions
2. Collections, Arrays
3. Generics
4. Layouts, Activities, Intent, Fragments
5. UI Thread, Threads and Runnables, Handler Class
6. Async Task, Anko's doAsync .
7. Shared Preferences
8. Broadcast Receivers(System Broadcast, Custom Broadcast).