VIJAYANAGARA SRI KRISHNADEVARAYA UNIVERSITY
Jnana Sagara Campus, Vinayakanagara, Cantonment,
BALLARI - 583 105.

Department of Studies in Computer Science

Programme Outcomes (POs):

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

PO1: Computer knowledge: Apply the knowledge of mathematics, science and engineering fundamentals to the solution of complex problems

PO2: Problem analysis: Identify, formulate, review research literature, and analyze complex problems reaching substantiated conclusions using principles of mathematics, natural sciences.

PO3: Design/development of solutions: Design solutions for complex problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO4: Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern IT tools to complex problems with an understanding of the limitations.

PO5: Environment and sustainability: Understand the impact of the professional solution in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO6: Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO7: Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO8: Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO9: Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Course Outcomes (COs):

I Semester

CS-HC 1.1: Digital Logic and Computer Design

Teaching: 4 hrs/week **Credits: 04**
Max Marks: 100 & total Hours: 52
Code: **CS-HC** 1.1 Univ Code:_____
Contact Hours: 4 hours per week Work load: 4 hours per week
Credit Points: 04
Evaluation: Continuous Internal Assessment - 30 marks
Semester and Examination - 70 marks

UNIT–I **8Hrs**

Digital computers and digital systems, binary numbers, number base conversion, octal and hexadecimal numbers, complements, binary codes, binary storage and registers, binary logic and integrated circuits.

UNIT–II **10Hrs**

Definition of Boolean algebra, basic theorems and properties of Boolean algebra, Boolean functions, canonical and standard forms, digital logic gates, IC digital logic families, simplification of Boolean functions, two, three and four variable maps, sum of products and product of sums simplification, NAND and NOR implementation, non degenerate forms, AND-OR-INVERT implementation, Don't-Care conditions, the tabulation method, determination and selection of prime-implicants.

UNIT–III **8Hrs**

Combinational circuit, design procedure, adders, subtractors, code conversion, analysis procedure, multilevel NAND and NOR circuits, exclusive-or and equivalence functions, binary parallel adder, decimal adder, magnitude comparators, decoders, multiplexers, Read-Only memory, Programmable Logic Array.

UNIT–IV **8Hrs**

Sequential circuit, flip-flops, analysis of clocked sequential circuits, flip-flop excitation tables, design procedure, design of counters, design with state equations.

UNIT–V **10Hrs**

Registers, shift registers, ripple counters, synchronous counters, timing sequences, the memory unit, examples of random access memory, interregister transfer, arithmetic, logic, and shift micro-operations, conditional control statements, fixed-point binary data, overflow, arithmetic shifts, decimal data, floating-point data, non-numeric data, instruction codes, design of simple counter.

UNIT–VI **8Hrs**

Processor organization, arithmetic logic unit, design of arithmetic logic unit, status register, design of shifter, processor unit, design of accumulator, control organization, microprogram control, control of processor unit, microprogram sequencer.

Title of the Course with Code: Digital Logic and Computer Design **CS-HC 1.1**

After completion of this course students will be able to

CO	**Statement**
CO1	Describe, Illustrate and analyze Combinational Logic circuits, Simplification of Algebraic Equations using Karnaugh Maps
CO2	Describe the working and Design of Decoders, Digital multiplexers, Adders and Subtractors, and Master-Slave Flip-Flops.
CO3	Design different synchronous and asynchronous sequential circuits and their applications
CO4	Analyze the working of ADC and DAC circuits and its applications

References:

1. Morris Mano M., Digital logic and Computer Design, PHI .
2. Floyd and Jain, Digital Fundamentals, 8/e, Pearson Education.
3. Alan B Marcovitz, Introduction to logic and Computer Design, McGraw Hill.
4. Ronald J. Tocci, Digital Systems: Principals and Applications, 8/e, Pearson Education.
5. Bartee J. C., Digital Computer Fundamentals, 6/e, TMH.
6. Herbert Taub and Donald Schilling, Digital Integrated Electronics, McGraw Hill International Edition .
7. Ramesh S. Gaonkar., Microprocessor Architecture, Programming, and Applications with the 7085, 4/e, Penram International Publishers.

CS-HC 1.2: Mathematical Foundation for Computer Science

Teaching: 4 hrs/week **Credits: 04**

Max Marks: 100 & Total Hours: 52

Code: **CS-HC** 1.2 Univ Code: _____

Contact Hours: 4 hours per week Work load: 4 hours per week

Credit Points: 04

Evaluation: Continuous Internal Assessment - 30 marks

Semester and Examination - 70 marks

Unit I **10Hrs**

Relations and Functions: Sets, sequences, matrices, mathematical structures, product sets and partitions, relations and digraphs, properties of relations, equivalence relations, operations on relations, transitive closure and Warshall's algorithm, functions, Permutation functions.

Unit II **10Hrs**

Combinatorics: permutations, combinations, Pigeonhole principle, recurrence relations, principle of Inclusion and Exclusion, generating functions.

Unit III **10Hrs**

Order Relations and Structures: Partially ordered set, lattices, finite Boolean algebras, functions on Boolean algebras, circuit designs.

Unit IV **10Hrs**

Graphs and Trees: Graphs, Euler paths and circuits, Hamiltonian paths and circuits, transport networks, matching problems; trees, labeled trees, tree searching, undirected trees, minimal spanning trees.

Unit V **12Hrs**

Groups and Coding: Semi groups, groups, coding of binary information and Error detection, decoding and error correction.

Title of the Course with Code: Mathematical Foundation for Computer Science **CS-HC** 1.2

After completion of this course students will be able to

CO	**Statement**
CO1	Understand sets, relations, functions and discrete structures.
CO2	Apply propositional logic and first order logic to solve problems.
CO3	Formulate and solve graph problems.

References:

1 Kolman ,Busby and Ros , Discrete Mathematical Structures , 4/e , Pearson Education.

2 Ralph P. Grimaldi, Discrete and Combinatorial Mathematics, 4/e , Pearson Education.

3 Purna Chandra Biswal, Discrete Mathematics and Graph Theory, PHI.

4 Trembley J.P. and Manohar R., Discrete Mathematical Structure with Application to Computer Science. TMH.

5 Kishore Trivedi, Probability and Statistics with Reliability, Queuing and Computer Science Applications, PHI.

CS-HC 1.3: Data Structures using C++

Teaching: 4 hrs/week **Credits: 04**

Max Marks: 100 & Total Hours: 52

Code: **CS-HC** 1.3 Univ Code: _____

Contact Hours: 4 hours per week Work load: 4 hours per week

Credit Points :04

Evaluation: Continuous Internal Assessment -30 marks

Semester and Examination -70 marks

UNIT-I **6Hrs**

Object oriented programming, concepts of OOP, advantages of OOP, ADT, a model for an ADT, algorithm efficiency, list searches-sequential and binary search algorithm, linear list concepts, linked list concepts, linked list algorithms, Processing a linked list, list applications, complex linked structures, C++ implementation, list ADT.

Unit II **12Hrs**

Stack definition, basic stack operations, stack linked list implementation, stack applications, C++ implementation, stack ADT implementation, stack ADT-array implementation, queue definition, queue operations, queue linked list design, queue applications, C++ implementation, queue ADT-linked list and array implementation.

Unit III **6Hrs**

Recursion, designing recursive algorithms, case study-factorial, Fibonacci numbers, towers of Hanoi, C++ implementation.

Unit IV **12Hrs**

Trees, basic tree concepts, binary trees, binary tree traversal, expression trees, general trees, Huffman code, binary search trees, AVL trees, AVL tree implementation, AVL ADT, heap definition, heap algorithms, m-way search trees, B-trees, lexical search tree, B-Tree ADT.

Unit V **8Hrs**

Sorting concepts, insertion sort, selection sort, exchange sort, external sorts.

Unit VI **8Hrs**

Graphs, graph operations, graph storage structures, graph algorithms, networks.

Title of the Course with Code: Data Structure using C++ **CS-HC** 1.3

After completion of this course students will be able to

CO	Statement
CO1	Design programs using a variety of data structures such as stacks, queues, binary trees, heaps, graphs.
CO2	Analyze and implement various kinds of searching and sorting techniques
CO3	Discuss the applications of various Data Structures

References:

1. Paul S. Wang, Standard C++ with Object Oriented Programming, Thomson Learning.
2. S. B. Lippman & J. Lajoie, C++ Primer, 3rd Edition, Addison Wesley.
3. B. A. Forouzon, R. F. Gilberge, Computer Science: A Structured Approach Using C++, Thomson Learning.
4. Herbert Schildt, C++-The Complete Reference, TMH.
5. R. F. Gilberg and B. A. Forouzan, Data Structures-A Pseudocode Approach with C++, Thomson Learning.
6. Mark A. Weiss, Data Structures and Algorithm Analysis in C++, 2/e, Pearson Education.
7. Langsam Yedidyah, Augenstein Moshe J., Tenenbaum Aaron M., Data Structures Using C and C++, 2/e,PHI/Pearson Education.
8. Samanta. D., Classic Data Structures, PHI.

CS-SC 1.4 (a): Operating System Principles

Teaching: 4 hrs./week **Credits: 04**

Max Marks: 100 & Total Hours: 52

Code : **CS-SC** (a)1.4 Univ Code : _____

Contact Hours : 4 hours per week Work load : 4 hours per week

Credit Points :04

Evaluation: Continuous Internal Assessment - 30 marks

Semester and Examination - 70 marks

UNIT-I **13 Hrs**

Introduction: Operating system structure, operations, overview of process management, memory management, storage management and protection and security; distributed systems, special purpose systems, computing environments.

System Structure: Operating system services under OS interface, system calls, system programs, operating system design and implementation, OS structure, virtual machines, system boot.

Unit II **13Hrs**

Process Management and Process Coordination-Synchronization and deadlocks: Process scheduling, operations on processes, inter process communication, communication in client server systems, multithreaded programming, scheduling criteria, scheduling algorithms, thread scheduling, algorithm service, Synchronization, the critical section problem, Peterson's solution, synchronization hardware, semaphores, classical problems of synchronization, monitors, synchronization examples, atomic transaction, deadlock characterization, methods of handling deadlocks, deadlock prevention and avoidance, deadlock detection, recovery from deadlock.

Unit III **13Hrs**

Memory Management: Swapping, contiguous memory allocation, paging, structure of page table, segmentation, example: the Intel Pentium, demand paging, copy-on-write, page replacement, allocation of frames, thrashing, memory-mapped files, allocating Kernel memory, examples. Storage Management-File System and Secondary storage structure: File concept, access methods, directory structure, File-System mounting, file sharing, protection, file-system structure and implementation, directory implementation, allocation methods, free-space management, efficiency and performance, NFS, example-The WAFL file system, disk Structure, disk attachment, disk scheduling, disk management, swap-space management, RAID structure, stable-storage implementation, tertiary storage structure.

Unit IV **13Hrs**

Case Study- Linux Internals: Linux User and programmer Interface, File system, process management, interprocess communication, Memory management, Understanding shells,shell programming.

Title of the Course with Code: Operating System Principles **CS-SC** (a)1.4

After completion of this course students will be able to

CO	Statement
CO1	Explain the core structure and functionality of operating system.
CO2	Discuss and analyze various inter process communication mechanisms.
CO3	Evaluate and analyze the different techniques for solving CPU scheduling problems.
CO4	Describe and Apply the knowledge of deadlock concepts to provide wide range of functionality to applications.
CO5	Identify and analyze the problems that occur in the design of OS based on knowledge gained through process synchronization techniques.
CO6	Analyze the performance of different memory management techniques and page replacement algorithms.

Reference:

1 Silberschartz A. and Galvin P., Operating System Concepts, 7/e, Addison Wesley.
2 Gary J. Nutt, Operating Systems, Addition-Wesley.
3 I. M. Flyn, A. Mclver McHoes., Understanding Operating Systems, Thomson Learning.
4 D. M. Dhamdhare, Operating Systems, Tata Mc.Graw-Hill.
5 Deitel H.M., An Introduction to Operating Systems, Addison Wesley.
6 Jack Dent, Tony Gaddis, Guide to UNIX using Linux, Thomson Learning.
7 Nicholas Wells, Guide to Linux installation and Administration, Thomson Learning.

CS-SC 1.4 (b): Problem Solving Technique using C

Teaching: 4 hrs./week **Credits: 04**

Max Marks: 100 & Total Hours: 52

Code : **CS-SC** (b)1.4 Univ Code : _____

Contact Hours : 4 hours per week Work load : 4 hours per week

Credit Points :04

Evaluation: Continuous Internal Assessment - 30 marks

Semester and Examination - 70 marks

UNIT - I **12 Hrs**

Introduction to Programming Concepts: Software, Classification of Software, Modular Programming, Structured Programming, Algorithms and Flowcharts, Writing algorithms and drawing flowcharts for simple exercises. Overview of C Language: History of C, Character set, C tokens, Identifiers, Keywords, structure of C program, executing a C program. Constants, variables, data types, declaration of variables, declaration of storage classes, assigning values to variables defining symbolic constants, declaring a variable as constant, declaring a variable as volatile, overflow and underflow of data, Operators in C, Hierarchy of Operators, Expressions, Type Conversions and Library Functions.

UNIT – II **10 Hrs**

Managing Input and Output Operations: The scanf() & printf() functions for input and output operations, reading a character, writing a character, (the getchar() & putchar() functions) , the address operator(&), formatted input and output using format specifiers, Writing simple complete C programs. Control Statements: Decision making with if statement, simple if statement, the if..else statement, nesting of if..else statements, the else..if ladder, the switch statement, the ?: operator, the goto statement, the break statement, programming examples. Loop Control Structures: The while statement, the do..while statement, the for statement, nested loops, jumps in loops, the continue statement, programming examples.

UNIT – III **10 Hrs**

Functions: Function Definition, prototyping, types of functions, passing arguments to functions, Nested Functions, Recursive functions. Arrays: Declaring and Initializing, One Dimensional Arrays, Two Dimensional Arrays, Multi Dimensional Arrays - Passing arrays to functions. Strings: Declaring and Initializing strings, Operations on

strings, Arrays of strings, passing strings to functions. Storage Classes - Automatic, External, Static and Register Variables.

UNIT – IV **10 Hrs**

Structures and Unions: Defining a structure, declaring structure variables, accessing structure members, structure initialization, copying and comparing structure variables, operations on individual members, array of structures, structures within structures, structures and functions, Unions, size of structures, bit fields, programming examples. Pointers: Understanding pointers, accessing the address space of a variable, declaring and initialization pointer variables, accessing a variable through its pointer, chain of pointers, pointer expressions, pointers and arrays, pointer and character strings, array of pointers, pointer as function arguments, functions returning pointers, pointers to functions, pointers and structures, programming examples

UNIT – V **10 Hrs**

File Management in C: Defining and opening a file, closing a file, input/output operations on files, error handling during I/O operations, random access files, command line arguments, programming examples. Dynamic Memory Allocation: Dynamic memory allocation, allocating a block of memory: malloc, allocating multiple blocks of memory: calloc, releasing the used space: Free, altering the size of a block: realloc, programming examples. The Preprocessor: Introduction, macro substitution, files inclusion, compiler control directives, ANSI additions, programming exercises.

Title of the Course with Code: Problem Solving Techniques using C **CS-SC** (b)1.4

After completion of this course students will be able to

CO	**Statement**
CO1	Illustrate and explain the basic computer concepts and programming principles of C language.
CO2	Develop C programs to solve simple mathematical and decision making problems.
CO3	Develop C programs to solve simple problems using looping constructs.
CO4	Develop C programs to demonstrate the applications of derived data types such as arrays, pointers, strings and functions.

Reference

1. *E. Balaguruswamy, "Programming In ANSI C", 4th Edition, TMH Publications, 2007.*
2. *Ashok N. Kamthane, "Programming with ANSI and Turbo C", Pearson Education, 2006.*
3. *Mahapatra, " Thinking In C ", PHI Publications, 1998.*

CS-HCP 1.5 (a): Practical – I: Digital Logic and Linux/Unix Lab.

Practical: 4 hrs./week **Credits: 04**

Max Marks: 100 & Total Hours: 52

Code: **CS-HCP** (a)1.5 Univ Code: _____

Contact Hours: 4 hours per week Work load: 4 hours per week

Credit Points: 04

Evaluation: Continuous Internal Assessment - 30 marks
Semester and Examination - 70 marks

Section I: Lab. Assignment shall be carried out based on the paper MSC 1.1 including the following:

- Realization of NOT, OR, AND, XOR, XNOR gates using universal gates
- Gray to Binary conversion & vice-versa.
- Code conversion between BCD and EXCESS-3
- ODD and even parity generation and checking.
- 4-bit comparator circuit
- Design of combinational circuit to drive seven-segment display
- Design of combinational circuits using multiplexer
- Adder/Subtractor circuits using Full-Adder using IC and/ or logic gates.
- BCD Adder circuit using IC and/ or logic gates
- Realization of RS, JK, and D flip flops using Universal logic gates
- Realization of Asynchronous up/down counter
- Realization of Synchronous Mod-N counter

Section II : Lab. Assignment shall be carried out to include the following features of Linux/UNIX:

- Basic commands, File system commands
- Process management, interprocess communication
- Search and sort tools, AWK tool, Shell programming, make tool, tar utility
- System administration.

Lab. Assignment shall be carried out to simulate the following OS features using c/c++

- cpu scheduling algorithms
- memory management scheme, demand paging scheme
- disk scheduling algorithms
- Interprocess communication

CS-HCP 1.5 (b): Practical – I: Problem Solving Technique Using C lab

Practical: 4 hrs./week **Credits: 04**

Max Marks: 100 & Total Hours: 52

Code: **CS-HCP** (b)1.5 Univ Code: _____

Contact Hours: 4 hours per week Work load: 4 hours per week

Credit Points: 04

Evaluation: Continuous Internal Assessment - 30 marks

Semester and Examination - 70 marks

Section I: Lab. Assignment shall be carried out based on the paper MSC 1.1 including the following:

- Realization of NOT, OR, AND, XOR, XNOR gates using universal gates
- Gray to Binary conversion & vice-versa.
- Code conversion between BCD and EXCESS-3
- ODD and even parity generation and checking.
- 4-bit comparator circuit
- Design of combinational circuit to drive seven-segment display
- Design of combinational circuits using multiplexer
- Adder/Subtractor circuits using Full-Adder using IC and/ or logic gates.
- BCD Adder circuit using IC and/ or logic gates
- Realization of RS, JK, and D flip flops using Universal logic gates
- Realization of Asynchronous up/down counter
- Realization of Synchronous Mod-N counter

Section II :

1. Write a C Program to demonstrate all the operators.
2. Write a C Program for electricity bill tacking different Categories of users, different slabs in each category.
3. Write a C Program to find check whether the given number is Prime or not.
4. Write a menu driven C Program to find the factorial of number (a) Without function (b) Using non-recursive function (c) Using Recursive Function.
5. Write a C Program to check the correctness of the date and compare two dates.
6. Write a C Program to find the sum of its individual digits repeatedly till the result is a single digit.
7. Write a program to enter integer number and find the largest and smallest digit of the number.
8. Write a program to read three digits +ve integer number 'n' and generate possible permutations of number using their digits.
9. Write a C Program to accept a text upto 50 words and perform following actions
    a) Count total vowels, constants, spaces, sentences and words with spaces.
    b) Program should erase more than one space between two successive words.

10. Write a C program to enter names of cities and display all the entered names alphabetically.
11. Write menu Driven C Program to calculate to calculate sin, cos and exponential series without using standard library function.
12. Write a C Program to accept array of elements in unsorted order, sort the array and search an element using binary search.
13. Write a C Program to add and multiply two matrices.
14. Write a C Program to display list of C program files and directories.
15. Write a program to use macros as an array and pointer.
16. Write a program to display the attributes of a file using dos interrupt.
17. Write a program to delete a file using dos interrupt.
18. Create user defined data type equivalent to int. Declare three variables of its type. Perform arithmetic operations using these variables.
19. Write a program to read a C program file and count the following in the complete program. a) Total number of statements b) Total number of included files c) Total number of brackets.
20. Write a program to display C Program files in current directory. The user should select one of the files. Convert the file contents in Capital and Display the same on the screen.
21. Write a program to interchange the contents of two files.
22. Write a program to change mouse cursor.

CS-HCP 1.6: Practical – II: C++ and Data Structures Lab.

Practical: 4 hrs./week **Credits: 04**

Max Marks: 100 & Total Hours: 52

Code: **CS-HCP** 1.6 Univ Code: _____

Contact Hours: 4 hours per week Work load: 4 hours per week

Credit Points: 04

Evaluation: Continuous Internal Assessment -30 marks

Semester and Examination -70 marks

Section I: Data Structure algorithms studied in paper MSC1.3 shall be implemented using C++. Assignments should include but not limited to-

- Linked lists: inserting, deleting, inverting a linked list
- Stacks and Queues: adding, deleting elements
- Circular Queue: Adding & deleting elements
- Evaluation of expressions
- Polynomial addition, Polynomial multiplication
- Sparse Matrices: Multiplication, addition.
- Recursive and Non recursive traversal of Trees
- Threaded binary tree traversal. AVL tree implementation
- Application of Trees.
- Application of sorting and searching algorithms.

II Semester

CS-HC 2.1: Design and Analysis of Algorithms

Teaching: 4 hrs./week **Credits: 04**

Max Marks: 100 & Total Hours: 52

Code: **CS-HC** 2.1 Univ Code: _____

Contact Hours: 4 hours per week Work load: 4 hours per week

Credit Points: 04

Evaluation: Continuous Internal Assessment - 30 marks

Semester and Examination - 70 marks

Unit I **5Hrs**

Notion of algorithm, Fundamentals of algorithmic problem solving, problem types, linear data structures, graphs, trees, sets and dictionaries.

Unit II 10Hrs Analysis of algorithm efficiency: Analysis frame-work, asymptotic notations and basic efficiency classes, mathematical analysis of non recursive and recursive algorithms, empirical analysis of algorithms.

Unit III **9Hrs**

Brute Force and Divide and Conquer: selection sort and bubble sort, sequential search and brute-force string matching, closest-pair and convex -hull problems, exhaustive search, merge sort, quick sort, binary search, binary tree traversals, Strassen's matrix multiplication.

Unit IV **10Hrs**

Decrease-and-Conquer and Transform-and-Conquer: Depth first search, Breadth First Search, topological sorting, balanced search trees, heap sort, Horner's rule.

Unit V **5Hrs**

Dynamic programming: Computing a Binomial coefficient, Warshall's and Floyd's algorithms, the Knapsack problem and memory functions.

Unit VI **5Hrs**

Greedy technique-Prim's algorithm, Dijkstra's algorithm, Huffman trees.

Unit VII **8Hrs**

The Fast Fourier Transform and its Applications: The discrete Fourier transform and its inverse, the Fast Fourier transform algorithm, the FFT using bit operations, products of polynomials, the Schonhage-Strassen integer-multiplication algorithm.

Title of the Course with Code: Design and Analysis of Algorithm **CS-HC** 2.1

After completion of this course students will be able to

CO	Statement
CO1	Identify the fundamental principles of algorithm analysis and design.
CO2	Analyze the complexity of a given algorithm
CO3	Apply design techniques such as divide-andconquer, decrease and conquer to solve a given problem
CO4	Apply the design techniques such as dynamic programming and greedy technique to solve a given problem

References:

1. Anany Levitin, The Design and Analysis of Algorithms, Pearson Education.
2. Aho A.V, Hopcroft J.E and Ullman, J.D., The Design and Analysis of Computer Algorithms, Addison – Wesley.
3. Ellis, Horwitz, Sartaj Sahani and S. Rajashekaran, Computer Algorithms, Galgotia Publications Pvt. Ltd.
4. David Harel, Algorithmics: The Spirit of Computing, Pearson Education.
5. Sara Baase, Computer Algorithms – An Introduction to Design and Analysis, Addison Wesley.

CS-HC 2.2: Database Management System

Teaching: 4 hrs/week **Credits: 04**
Max Marks: 100 & Total Hours: 52

Code: **CS-HC** 2.2 Univ Code: _____
Contact Hours: 4 hours per week Work load: 4 hours per week
Credit Points: 04
Evaluation: Continuous Internal Assessment -30 marks
Semester and Examination - 70 marks

Unit I **4Hrs**

Introduction: Database, characteristics of database approach, database users, advantages of database systems.

Unit II **8Hrs**

Database System Concepts and Architecture: Data models, schemas and instances, the three schema architecture, data independence, DBMS languages and interfaces, DBMS component modules and database system utilities, overview of Relational Data Base Management Systems, data modeling using Entity-Relationship Model.

Unit III **10Hrs**

The Relational Data Model: Relational models concepts, relational constraints and relational database schemas, update operations and dealing with constraint violations, relational algebra, relational calculus, and relational database design by ER to Relational Mapping.

Unit IV **8Hrs**

Relational Database Manipulation- SQL: Data definition in SQL, basic data retrieval, condition specification, arithmetic and aggregate operators, SQL join, set manipulation, Categorization, updates, views, views and updates.

Unit V **10Hrs**

Relational Database Design: Anomalies in a database-A consequence of bad design, functional dependencies, Normal forms based on primary keys, general definitions of second and third normal forms, Boyce-Codd normal form, relational database design algorithms, multivalued dependencies and fourth normal form, join dependencies and fifth normal form.

Unit VI **12Hrs**

System Implementation Techniques: Database System Architecture and the System Catalog, query processing and optimization, transaction processing concepts, concurrency control techniques, database recovery techniques, database security and Authorization.

Title of the Course with Code: Database Management System **CS-HC** 2.2

After completion of this course students will be able to

CO	Statement
CO1	Describe the fundamentals of database technologies, Design an ER diagram and transform it to a relational model for a given database specification.
CO2	Discuss the relational model concepts and Design relational algebraic expressions for queries.
CO3	Explain the various concepts of SQL and Design SQL queries to perform CRUD (Create, Retrieve, Update and delete) operations on database.
CO4	Discuss the database design concepts such as functional dependency and solve the problems on minimal set, equivalence set.
CO5	Discuss the database design concepts such as Normalization, Relational decomposition and concepts of transaction processing. Apply the normalization techniques to improve database design.

References:

1. Henry F. Korth and Silberschatz Abraham, Database System Concepts, Mc.Graw Hill.
2. Elmasri and Navathe, Fundamentals of Database Systems, Pearson Education.
3. Bipin C. Desai, An Introduction to Database Systems, Galgotia Publications.
4. Date, C. J., An Introduction to Database Systems, Addison-Wesley.
5. Kroenke David M., Database Processing Fundamentals, Design, and Implementation, PHI.
6. Shah, Database Systems Using Oracle-A simplified guide to SQL and PL/SQL, PHI.

CS-SC 2.3 (a): System Software

Teaching: 4 hrs./week **Credits: 04**

Max Marks: 100 & Total Hours: 52

Code: **CS-SC** (a)2.3 Univ Code: _____

Contact Hours: 4 hours per week Work load: 4 hours per week

Credit Points: 04

Evaluation: Continuous Internal Assessment - 30 marks
Semester and Examination - 70 marks

UNIT –I **10 Hrs**

Introduction: System software and machine architecture, traditional (CISC) machines, RISC machines.

Unit II **10Hrs**

Assemblers: Basic assembler functions, machine dependent and machine independent assembler features, one-pass assemblers, multipass assemblers, MASM assembler, SPARC assembler.

Unit III **10Hrs**

Loaders and Linkers: Basic loader functions, machine dependent and machine independent loader features, linkage editors, dynamic linking, bootstrap loaders.

Unit IV **10Hrs**

Macro Processors: Basic macro processor functions, machine dependent and machine independent macro processor features, macro processor design options.

Unit V **12Hrs**

Compilers: Basic compiler functions, machine-dependent compiler features, machine-independent compiler features, compiler design options the YACC compiler-compiler.

Title of the Course with Code: System Software CS-SC (a)2.3

After completion of this course students will be able to

CO	Statement
CO1	Understand the concepts of SIC machine architecture, CISC and RISC machines
CO2	Explain the various concepts of machine dependent and machine independent assembler features and its types.
CO3	Discuss on the concepts of loaders, linkers and macro processors features.
CO4	Understand the concepts of compiler functions and its design options.

References:

1. Leland L. Black, System Software, Pearson Education.
2. A.V. Aho, R. Semi, J.D. Ullman, Compilers - Principles, techniques and tools, Pearson Education.
3. D.M. Dhamdhere, Systems Programming and Operating Systems, Tata McGraw Hill.

CS-SC 2.3 (b): Object Oriented Analysis and Design using UML

Teaching: 4 hrs/week **Credits: 04**

Max Marks: 100 & Total Hours: 52

Code: **CS-SC** (b)2.3 Univ Code: _____

Contact Hours: 4 hours per week Work load: 4 hours per week

Credit Points: 04

Evaluation: Continuous Internal Assessment - 30 marks

Semester and Examination - 70 marks

UNIT – I **12 Hrs**

Introduction: An overview - Object basics - Object state and properties, Behavior, Methods, Messages. Object Oriented system development life cycle, Benefits of OO Methodology. Overview of Prominent OO Methodologies: The Rumbaugh OMT, The Booch methodology, Jacobson's OOSE methodologies, Unified Process, Introduction to UML, Important views & diagram to be modelled for system by UML. Factional View (models): Use case diagram - Requirement Capture with Use case - Building blocks of Use Case diagram - actors, use case guidelines for use case models - Relationships between use cases - extend, include, generalize. Activity diagram - Elements of Activity Diagram - Action state, Activity state, Object, node, Control and Object flow, Transition (Fork, Merge, Join) - Guidelines for Creating Activity Diagrams - Activity Diagram - Action Decomposition (Rake) - Partition - Swim Lane.

UNIT – II **10 Hrs**

Static structural view (Models): Classes, values and attributes, operations and methods, responsibilities for classes, abstract classes, access specification (visibility of attributes and operations). Relationships among classes: Associations, Dependencies. Inheritance - Generalizations, Aggregation. Adornments on Association: association names, association classes, qualified association, n-ary associations, ternary and reflexive association. Dependency relationships among classes, notations. Notes in class diagram, Extension mechanisms, Metadata, Refinements, Derived, data, constraint, stereotypes, Package & interface notation. Object diagram notations and modeling, relations among objects (links).

UNIT – III **10 Hrs**

Class Modeling and Design Approaches: Three approaches for identifying classes - using Noun phrases, Abstraction, Use Case Diagram - Comparison of approaches -

Using combination of approaches - Flexibility guidelines for class diagram: Cohesion, Coupling, Forms of coupling (identity, representational, subclass, inheritance), class Generalization, class specialization versus aggregation. Behavioral (Dynamic structural view): State diagram - State Diagram Notations, events (signal events, change events, Time events) - State Diagram states (composite states, parallel states, History states), transition and condition, state diagram behaviour (activity effect, do-activity, entry and exit activity), completion transition, sending signals.

UNIT – IV **10 Hrs**

Interaction diagrams: Sequence diagram - Sequence diagram notations and examples, iterations, conditional messaging, branching, object creation and destruction, time constraints, origin of links, Activations in sequence diagram - Collaboration diagram - Collaboration diagram notations and examples, iterations, conditional messaging, branching, object creation and destruction, time constraints, origin of links, activations in sequence diagram. Approaches for developing dynamic systems: Top - down approach for dynamic systems - Bottom - up approach for dynamic systems - Flexibility Guidelines for Behavioral Design - guidelines for allocating and designing behaviors that lead to more flexible design.

UNIT – V **10 Hrs**

Architectural view: Logical architecture: dependency, class visibility, sub systems - Hardware architecture: deployment diagram notations, nodes, object migration between node - Process architecture: what are process and threads and their notations in UML, object synchronization, invocation schemes for threads (UML notations for different types of invocations). Implementation architecture: component diagram notations and examples. Reuse: Libraries, Frame works components and Patterns: Reuse of classes, Reuse of components, Reuse of frameworks, black box framework, white box frame, Reuse of patterns: Architectural pattern and Design pattern.

Title of the Course with Code: Object Oriented Analysis and Design using UML CS-SC (b)2.3

After completion of this course students will be able to

CO	**Statement**
CO1	Apply the fundamental knowledge of object oriented software development, methodologies, UML language, and design patterns to the solution of complex problems.

CO2	Analyze, formulate and review and justify a case study to identify classes, attributes, methods and relationships among them in the solutions.
CO3	Design UML models for a given case study using object oriented software development methodologies.

Reference

1. *Charles Richter, "Designing Flexible Object Oriented systems with UML"*
2. *Jackson, Burd Thomson, "Object Oriented Analysis & Design",*
3. *James Rumbaugh. Micheal Blaha, Object oriented Modeling and Design with UML.*
4. *Grady Booch, James Rumbaugh, Ivar Jacobson., "The Unified Modeling Language User Guide", Pearson Education.*
5. *James Rumbaugh, "Object Oriented Modeling and Design"*
6. *Joseph Schmuilers, "Teach Yourself UML in 24 Hours"*

Mike O'Docherty, "Object-Oriented Analysis and Design: using UML", Wiley Publication

CS-OE 2.4 (b): E- Governance

Teaching: 4 hrs/week **Credits: 04**

Max Marks: 100 & Total Hours: 52 Cont. Assessment.30

Code: **CS-HC** (b) 2.4 Univ Code: _____

Contact Hours: 4 hours per week Work load: 4 hours per week

Credit Points: 04

Evaluation: Continuous Internal Assessment -30 marks

Semester and Examination -70 marks

UNIT – I **12 Hrs**

Introduction to e- Governance, Different Stages of e-Governance, Advantages, Problems and Challenges of e-Governance, National Statues, International Status, Securities in e-Governance.

UNIT – II **10 Hrs**

National e-Governance Plan, Government of India guidelines for websites, W3C guidelines, web 2.0, web 3.0

UNIT – III **10 Hrs**

Different UN Survey on e-Governance, UN Survey on e-Governance – 2014, e-Government Act, 2002, Aadhaar Bill, 2016, II Administrative Reforms Committee Report 11, Digital India Programme, IT Act, 2008 Section 1 to 11A, Section 43 and 66

UNIT – IV **10 Hrs**

Workflow Management in e-Governance, Digital Divide, Mechanism to handle Digital Divide, Bridge the digital divide, M-Governance, e-Learning, Role of Social Media in e-Governance, Big data Analytics in e-Governance, Semantic web Analytics.

UNIT – V **10 Hrs**

Case Study: Election Commission, Indian Railway Reservation, Addhar – UID, Income Tax, SAKALA, Bhoomi, e-Commission, CET admission, Centralized Admission, Student Scholarship Management.

Title of the Course with Code: e-Governance CS-HC (b) 2.4

After completion of this course students will be able to

CO	**Statement**
CO1	Understand the concept of e-governance, and the associated benefits and drawbacks
CO2	Analyze the National e-Governance plan

CO3	Understand the workflow management in eGovernance.
CO4	Case study on some e-Governance bodies

Reference

1. *Mishra D.S (2007). E-Governance as reform strategy for combating corruption in delivery of public services. Indian Journal of Public Administration. LIII (3).*
2. *Bhogle Srinivas (2009). E-Governance. Selected Readings on Information Technology Management: Contemporary Issues ed. George Kelley. Information Science Reference, New York.*
3. *Bhuiyan H Shahjahan (2011). Modernizing Bangladesh public administration through e-governance: Benefits and challenges. 28, 54-65.*
4. *The World Wide Web Consortium (2008). Web Content Accessibility Guidelines (WCAG) 2.0. Downloaded on 10th January, 2012 from http://www.w3.org/*
5. *Government of India (2009). Guidelines for Indian Government websites. Downloaded on 15th January, 2012 from http://darpg.nic.in/*
6. *e-Government Act (2002). https://www.gpo.gov/fdsys/pkg/PLAW-107publ347/pdf/PLAW-107publ347.pdf*
7. *Digital India Programme. http://www.digitalindia.gov.in/*
8. *Information Technology Act, 2008. http://www.dot.gov.in/act-rules/information-technology-act-2000*
9. *Second Administrative Reforms Committee Report. Report 11: Promoting e-Governance: The SMART way Forwardhttp://arc.gov.in/*

CS-HCP 2.5: Practical- I: Algorithms and DBMS Lab.

Practical: 4 hrs. /week **Credits: 04**

Max Marks: 100 & Total Hours: 52

Code: **CS-HCP** 2.5 Univ Code: _____

Contact Hours: 4 hours per week Work load: 4 hours per week

Credit Points: 04

Evaluation: Continuous Internal Assessment -30 marks

Semester and Examination - 70 marks

Section-I: To understand the design and analysis of algorithms, following assignments shall be implemented using C/C++.

1. Divide-And-Conquer algorithms for searching and sorting.
2. Strassen's matrix multiplication
3. Dynamic programming: Warshall's algorithm and Dijkstra's algorithm
4. Greedy algorithm
5. FFT
6. Integer multiplication

Section-II: Lab. Assignment shall be carried out to include the following:

- SQL: Data definition in SQL, basic data retrieval, condition specification, arithmetic and aggregate operators, SQL join, set manipulation, categorization, updates, views, views and updates.
- Introduction to PL/SQL programming
- The student is to develop a logical and physical database design for the given problem.

  The logical design performs the following tasks: 1) Map the ER/EER diagrams to a relational schema. Be sure to underline all primary keys, include all

  necessary foreign keys and indicate referential integrity constraints. 2) Identify the functional dependencies in each relation, 3) Normalize to the highest normal form possible.

  - Perform physical design based above logical design using Oracle/MYSQL on Windows platform or MySQL/PostgreSQL on Linux platform.
  - Perform DML and DDL using all possible SQL commands and with the help anyone host languages like C, C++, VB etc (ie embedded SQL).
  - Perform DML and DLL using PL/SQL and PL/pgSQL for the above problem.

Title of the Course with Code: Algorithm and DBMS Lab CS-HCP 2.5

After completion of this course students will be able to

CO	**Statement**
CO1	Apply the concepts of divide-and-conquer, decrease and conquer for a given problem.
CO2	Understand and design the concepts of dynamic programming for a problem.
CO3	Design and implement database schema.
CO4	Design the queries using DDL, DML, DCL and TCL commands.

CS-HCP 2.6: Practical -II: Visual Programming Lab.

Practical: 4 hrs./week **Credits: 04**

Max Marks: 100 & Total Hours: 52

Code: **CS-HCP** 2.6 Univ Code: _____

Contact Hours: 4 hours per week Work load: 4 hours per week

Credit Points: 04

Evaluation: Continuous Internal Assessment -30 marks

Semester and Examination -70 marks

Assignments related to VB/VB.NET language shall be carried out including the following features:

- Decision and iterative constructs
- Procedures, functions and exceptional handling
- Arrays, enumeration and structure
- Working with forms, GUI interface with windows forms and designing menus
- Objects and classes
- Overloading, inheritance, over riding
- Interfaces, namespaces and collections
- Events and delegates
- Multithreading and garbage collection
- Database programming

Components and assemblies

Title of the Course with Code: Visual Programming Lab CS-HCP 2.6

After completion of this course students will be able to

CO	**Statement**
CO1	Apply the concepts of VB language for a problem.
CO2	Design and develop programs to implement procedures, functions, exception handling.
CO3	Design and develop programs to create the window forms, menus.
CO4	Design and develop programs to create the window forms, menus.

III Semester

CS-HC 3.1: Programming in JAVA

Teaching: 4 hrs/week **Credits: 04**

Max Marks: 100 & Total Hours: 52

Code: **CS-HC** 3.1 Univ Code: _____

Contact Hour: 4 hours per week Work load: 4 hours per week

Credit Points: 04

Evaluation: Continuous Internal Assessment -30 marks

Semester and Examination -70 marks

UNIT-I **10Hrs**

Basics of JAVA, Applications and Applets, using the tools in JDK, javadoc, java, jdb etc.

JAVA Language- keywords, constants, variables and Data Types. Operators and Expressions, Decision making, branching and Looping, Labeled Loops Statement, Jump statements: Break, Continue, and Return. Arrays and Strings-Creating an Arrays, one and two Dimension Arrays, String Array, String and String Buffer Classes, Wrapper Classes.

Unit II **10Hrs**

Classes, Objects and Methods Defining a class, adding variables and Methods, creating objects constructors, class inheritance, Basics types, using super, multi level hierarchy, abstract and final classes, object class, packages and interfaces, Access protection, Extending interfaces, packages. Exception Handling, Fundamentals exception types, uncaught exceptions, throws, throw, try -catch, final, built in exceptions, creating your own exceptions.

Unit III **6Hrs**

Applet Programming - Creating and executing Java applets, inserting applets in a web page, AWT Classes, Event Handling & Swing Classes.

Unit IV **8Hrs**

Multithreading Fundamentals, Java Thread model: priorities, synchronization, messaging, thread class, Runnable interface, Interthread communication, suspending, resuming and stopping threads.

Unit V **10Hrs**

Input/Output -Basics, Streams, Byte and Character streams, predefined streams, reading and writing from console and files .Using standard Java Packages (lang,util,io) Networking -Basics, networking classes and interfaces, using java.net package, doing TCP/IP and Datagram programming.

Unit VI **8Hrs**

JDBC -Setting the JDBC connectivity with a backend database. RMI -Two tier and Multitier Architecture, Object serialization, RMI Fundamentals, Programming using Java RMI Classes and interfaces. Servlets-Background, Life Cycle, Java Servlet Development kit, Servlet API, Handling HTTP Requests and responsing, Using Cookies, Session Tracking and security issues.

Title of the Course with Code: Programming in Java CS-HC 3.1

After completion of this course students will be able to

CO	**Statement**
CO1	Identify classes, objects, members of a class and the relationships among them needed for a specific problem.
CO2	Write java program using threads, event handling and input output utilities. Develop programs using the Java Collection API as well as the Java standard class library
CO3	Demonstrate the ability to use Threads and synchronization in java.
CO4	Explain and write input – output programming in java and applications using Applets.

References:

1. Patrick Naughton And Herbert Schildt, Java The Complete Reference, TMH Publication .
2. Cay S. Horstmann and Gary Cornell, Core JAVA 2, Volume-I, 7/e, Pearson Education.
3. Cay S. Horstmann and Gary Cornell, Core JAVA 2, Volume-II, 7/e, Pearson Education.
4. Bruce Eckel, Thinking in Java, 3/e, Prentice Hall.
5. Bill Shannon, Mark Hapner, Vlada Matena, James Davidson, Eduardo Pelegri-Llopart, Larry Cable, Java 2 Platform Enterprise Edition, Platform and Component Specifications , Addision Wesley.
6. Partrick Naughton, Herbert Schidlt, JAVA 2 -The Complete Reference, Tata McGraw Hill.

CS-HC 3.2: Data Communications and Computer Networks

Teaching: 4 hrs/week **Credits: 04**

Max Marks: 100 & Total Hours: 52

Code: **CS-HC** 3.2 Univ Code: _____

Contact Hours: 4 hours per week Work load: 4 hours per week

Credit Points: 04

Evaluation: Continuous Internal Assessment -30 marks

Semester and Examination -70 marks

UNIT I **06Hrs**

Data Communications:, Network Components and Categories, types of Connections, n Topologies –Protocols, ISO / OSI model, Transmission Media, Coaxial Cable, Fiber Optics, Line Coding.

UNIT II **12Hrs**

Data Link Layer: Error detection and correction, Parity, LRC, CRC, Hamming code, low Control and Error control, stop and wait, go back-N ARQ, selective repeat ARQ- sliding window, HDLC., LAN, Ethernet IEEE 702.3, IEEE 702.4, IEEE 702.5, IEEE 702.11, FDDI, SONET, Bridges.

UNIT III **06Hrs**

Network Layer: Internetworks, Packet Switching and Datagram approach, IP addressing methods, Subnetting, Routing, Distance Vector Routing, Link State Routing, Routers.

UNIT IV **10Hrs**

Transport Layer: Multiplexing, Demultiplexing, Sockets, User Datagram Protocol (UDP), Transmission Control Protocol (TCP), Congestion Control, Quality of services (QOS).

UNIT V **10Hrs**

Application Layer: Domain Name Space (DNS), SMTP, FTP, HTTP – WWW

UNIT VI **8Hrs**

Security: Cryptography, network security, security in Internet.

Title of the Course with Code: Data Communication & Computer Network CS-HC 3.2

After completion of this course students will be able to

CO	Statement
CO1	Apply basics of data communication and its components to understand computer networks technology. Enumerate the layers of TCP/IP and explain the functions of each layer.
CO2	Experiment with error detection and correction techniques and explain various

	transmission media
CO3	Analyze various data link and network layer services.
CO4	Analyze various transport and application layer services
CO5	Discuss the various security oriented protocols in internet

References

1. Behrouz A Forouzan, Data Communications and Networking, Tata McGraw-Hill.
2. William A. Shay, Understanding Communications and Networks, Thomson Learning.
3. William Stallings, Data and Computer Communications, 7/e, Pearson Education.
4. Stevens et. al., Unix network programming-The sockets and networking API, Vol. 1/ 3/e, PHI.
5. Stevens et. al., Unix network programming-Interprocess Communication, Vol. 2, 2/e, PHI.
6. Ames Chellis Charles Perkins, Matthew Strebe, Networking Essentials:Study Guide MCSE, Second Edition, BPB Publications.
7. Douglas E. Comer, Internetworking with TCP/IP, Vol. I- Principles, Protocols, and Architecture, 3/e, PHI.
8. Stevens W.R., UNIX Network Programming, Vol. I and Vol II, 2/e, PHI

CS-SC 3.3 (a): Computer Graphics

Teaching: 4 hrs/week **Credits: 04**

Max Marks: 100 & Total Hours: 52

Code: **CS-SC** 3.3 Univ Code: _____

Contact Hours: 4 hours per week Work load: 4 hours per week

Credit Points: 04

Evaluation: Continuous Internal Assessment -30 marks

Semester and Examination -70 marks

UNIT-I **6Hrs**

Introduction to computer graphics, programming in sample raster graphics package (SRGP), graphics hardware.

Unit II **10Hrs**

Basic raster graphics algorithms for drawing 2D primitives,: scan converting lines, circles and ellipses. Filling rectangles, polygons and ellipse arcs; pattern filling, thick primitives, clipping lines, circles, ellipse and polygons.

Unit III **10Hrs**

Geometrical transformations: 2D transformations, homogeneous coordinates, matrix representation of 2D transformations, window-to-viewport transformation, 3D-transformations, composition of 2D and 3D transformations, viewing in 3D.

Unit IV **8Hrs**

Representing curves and surfaces: Polygon meshes, parametric cubic curves, parametric bicubic surfaces.

Unit V **8Hrs**

Solid modeling, achromatic and colored light, Dialog design and user interface software.

Unit V1 **10Hrs**

Visible surface determination: Functions of two variables, techniques for efficient visible surface algorithms, algorithms for visible line determination, the z-buffer algorithm, list-priority algorithm, scan-line algorithm, area-subdivision algorithm, algorithm for octrees and curved surfaces, visible surface ray tracing.

Title of the Course with Code: Computer Graphics CS-SC 3.3

After completion of this course students will be able to

CO	**Statement**
CO1	Apply basics of graphics to create interactive applications using one or more graphics application programming interfaces.
CO2	Design and implement programs to demonstrate 2D image processing techniques.
CO3	Demonstrate the 2D and 3D transformations, creating polygons, solid modeling.

References:

1. James D. Foley, Andres Van Dam, Steven K. Feiner, and John F. Hughies, Computer Graphics- Principles and Practice, 2/e, Pearson Education (3006).

2. Donald Hearn and M. Pauline Baker, Computer Graphics-C version, 2/e, Pearson Education.

CS-SC 3.3SC (b): Distributed Operating System

Teaching: 4 hrs/week **Credits: 04**

Max Marks: 100 & Total Hours: 52

Code: **CS-SC** 3.3 Univ Code: _____

Contact Hours: 4 hours per week Work load: 4 hours per week

Credit Points: 04

Evaluation: Continuous Internal Assessment -30 marks

Semester and Examination -70 marks

UNIT – I **12 Hrs**

Fundamentals: What is Distributed Computing Systems? Evolution of Distributed Computing System; Distributed Computing System Models; What is Distributed Operating System? Issues in Designing a Distributed Operating System; Introduction to Distributed Computing Environment (DCE). Message Passing: Introduction, Desirable features of a Good Message Passing System, Issues in PC by Message Passing, Synchronization, Buffering, Multidatagram Messages, Encoding and Decoding of Message Data, Process Addressing, Failure Handling, Group Communication, Case Study: 4.3 BSD UNIX IPC Mechanism.

UNIT – II **10 Hrs**

Remote Procedure Calls: Introduction, The RPC Model, Transparency of RPC, Implementing RPC Mechanism, Stub Generation, RPC Messages, Marshaling Arguments and Results, Server Management, Parameter-Passing Semantics, Call Semantics, Communication Protocols for RPCs, Complicated RPCs, Client-Server Binding, Exception Handling, Security, Some Special Types of RPCs, RPC in Heterogeneous Environments, Lightweight RPC, Optimization for Better Performance, Case Studies: Sun RPC.

UNIT – III **10 Hrs**

Distributed Shared Memory: Introduction, General Architecture of DSM Systems, Design and Implementation Issues of DSM, Granularity, Structure of Shared Memory Space, Consistency Models, Replacement Strategy, Thrashing, Other approaches to DSM, Heterogeneous DSM, Advantages of DSM. Synchronization: Introduction, Clock Synchronization, Event Ordering, Mutual Exclusion, Dead Lock, Election Algorithms.

UNIT – IV **10 Hrs**

Resource Management: Introduction, Desirable Features of a Good Global Scheduling Algorithm, Task Assignment Approach, Load – Balancing Approach, Load – Sharing Approach Process Management: Introduction, Process Migration, Threads.

UNIT – V **10 Hrs**

Distributed File Systems: Introduction, Desirable Features of a Good Distributed File System, File models, File– Accessing Models, File – Sharing Semantics, File – Caching Schemes, File Replication, Fault Tolerance, Atomic Transactions, Design Principles.

Title of the Course with Code: Distributed Operating System CS-SC 3.3

After completion of this course students will be able to

CO	**Statement**
CO1	Understand the concepts of Distributing computing environment.
CO2	Discuss the message passing mechanism and synchronization concepts in distributed computing environment.
CO3	Explain the concept of Remote Procedure Call and its implementation
CO4	Discuss the concepts of Distributed Shared memory, Resource Management and Distributed File System

Reference

1. *Pradeep. K. Sinha: Distributed Operating Systems: Concepts and Design, PHI, 2007.*
2. *Andrew S. Tanenbaum: Distributed Operating Systems, Pearson Education, 2013.*

CS-OE 3.4OE (a): Information Technology

Teaching: 4 hrs./week **Credits: 04**

Max Marks: 100 & Total Hours: 52

Code: **CS-OE** 3.4 Univ Code: _____

Contact Hours: 4 hours per week Work load: 4 hours per week

Credit Points: 04

Evaluation: Continuous Internal Assessment -30 marks

Semester and Examination -70 marks

UNIT I **10Hrs**

Introduction: Growth of computer networking, Complexity in network system, Motivation and Tools: Resource sharing, Growth of the internet, probing the internet, interpreting the ping response, tracing a route. Transmission Media: Copper wires, glass fibers.

UNIT II **12Hrs**

Communications: Introduction, the need for asynchronous communications, using electric current to send bits, standards for communication, baud rate, Framing and errors, Half and Full duplex asynchronous communication, the effect of noise on communication. Long distance Communication: Sending signals across long distances, Modem hardware used for Modulations and Demodulation, Leased analog data circuits, optical, radio frequency and dialup Modems, carrier frequencies and Multiplexing, baseband and broadband technologies, wave length division multiplexing, spread spectrum, time division multiplexing.

UNIT III **12Hrs**

Computer Networks: Definition, network types, network topology, network devices, OSI model, TCP/IP model, Local Area Network (LAN), Wide Area Network (WAN), Internetworking, IP addressing methods: IP addressing scheme, IP address hierarchy, classes of IP address & dotted decimal notation, addressing example, special IP address

UNIT III **10Hrs**

Internet Evolution, Basic Internet Terminology, Internet Essentials, Internet Services – USENET, TELNET, FTP, E-mail, HTTP, IRC, WORLD WIDE WEB.

UNIT IV **8Hrs**

Search Engines: Popular search engines, how to register a web site on internet, Blogs, Overview of HTML.

Title of the Course with Code: Information Technology : **CS-OE** 3.4

After completion of this course students will be able to

CO	Statement
CO1	Understand the concepts of basic networking concepts, different transmission media.
CO2	Enumerate the layers of OSI and TCP/IP and explain the functions of each layer
CO3	Discuss on various IP addressing methods, internet services and search engines.

References

1. Douglas E Comer, Internetworking with TCP/IP, Vol. I-Principles,Protocols,& Architecture,3/e,PHI.
2. V. Rajaraman, Introduction to Information Technology, PHI
3. P. K. Singh, Introduction to Computer Networks, V. K. Publications, New Delhi
4. Rachna Sharma, Computer Networks, University Science Press, Laxmi Publications.
5. Jesse Feiler, Managing the Web Based Enterprise, Morgan Kaufmann
6. Internet and Web Design, DOEACC 'O' level, Firewall Media.
7. Chuck Musciano & Bill Kennedy, HTML & XHTML, SPD.
8. Hossien Bidgoli, Elcetronic Commerce- Principles and Practice, Academic Press.
9. Efraim Turban, David King, Danis, Jae Lee, Electronic Commerce, Prentice Hall.
10. S.Jaiswal , Doing Business on the Internet : E – Commerce, Galgotia Pub.
11. Thomas A. Powell, the Complete Reference HTML.

IV Semester

CS-HC 4.1: Internetworking and Web Design

Teaching: 4 hrs/week **Credits: 04**

Max Marks: 100 & Total Hours: 52 Cont. Assessments. 30

Code: **CS-HC** 4.1 Univ Code: _____

Contact Hours: 4 hours per week Work load: 4 hours per week

Credit Points: 04

Evaluation: Continuous Internal Assessment -30 marks

Semester and Examination -70 marks

Unit I **10Hrs**

Internetworking: Motivation, physical connection with routers, internet architecture, concept of Universal service, virtual networks, protocols for internetworking, layering and TCP/IP protocols.

Internet protocol addresses: IP addressing scheme, IP address hierarchy, classes of IP addresses and dotted decimal notation, addressing example, special IP addresses. address resolution- ARP, ARP message delivery and format IP datagrams and datagram forwarding, IP encapsulation, fragmentation and reassembly.

Unit II **6Hrs**

IPv6: Characterization of features in IPv6, IPv6 datagram format, IPv6 base header format, fragmentation reassembly and path MTU, IPv6 addressing, IPv6 colon hexadecimal notation.

An error reporting mechanism (ICMP), TCP.

Unit III **12Hrs**

Web page building blocks, basic (X)HTML structure, basic (X)HTML formatting, images, links, style sheet building blocks, formatting with styles, dynamic effects with styles, layout with styles, list, tables, forms, video, audio, and other multimedia. Testing and debugging web pages.

Unit IV **10Hrs**

XML: What is XML?, limitations of values, XSL, DTD, XML schema, practice-AJAX, RSS, and SOAP. HTML, an XML sample, elements, attributes and name spaces, W3C recommendations, XML In

Unit V **14Hrs**

JAVA Script: The JAVA Script programming language, creating JavaScript source file, hiding JavaScript from incompatible browsers, variables, functions, objects and events, data types and operators, decision making with control structures and statements, windows and frames, working with forms in JavaScript, using JavaScript with CSS styles, cookies and security, introduction to document object model, debugging JavaScript, server side JavaScript, database connectivity, working with Java Applets and embedded data.

Title of the Course with Code: Internetworking & Web Design CS-HC 4.1

After completion of this course students will be able to

CO	**Statement**
CO1	Recall programming skills on internet based applications
CO2	Design and develop sophisticated web sites and applications.
CO3	Compare web projects developed with traditional projects
CO4	Critique procedures of internet programming
CO5	Implement the important HTML tags for designing static pages and separate design from content using Cascading Style sheet.

References:

1. Douglas E Comer, Computer Networks and Internet, Pearson Education.
2. Kevin Howard Goldberg, XML- Visual quick start guide, Peachpit Press.
3. David Hunter, Jeff Rafter, Joe, Eric, Danny, John, Andrew, Linda, Beginning XML, WROX publications.
4. Elizabeth Castro, HTML, XHTML, and CSS, Peachpit Press.
5. Deitel, Deitel, and Nieto, Internet & World Wide Web-How to Program, PHI.
6. Don Gosselin, JavaScript, Web Warrior Series,3/e, Thomson Learning.
7. Douglas E. Comer, Internetworking With TCP/IP, Vol. II: Design, Implementation, And Internals, 3/E, PHI.
8. Paul Wilton and Jeremy McPeak, Beginning Java Script, Wrox Publications.
9. Karl Barksdale, E. Turner, HTML, JavaScript, and Advanced Internet Technologies, Web Warrior Series,3/e, Thomson Learning.

CS-HC 4.2: Software Engineering

Teaching: 4 hrs./week **Credits: 04**

Max Marks: 100 & Total Hours: 52

Code: **CS-HC** 4.2 Univ Code: _____

Contact Hours: 4 hours per week Work load: 4 hours per week

Credit Points: 04

Evaluation: Continuous Internal Assessment -30 marks

Semester and Examination -70 marks

UNIT-I **8Hrs**

Introduction: Product and Process: Evolving role of software, software characteristic and components, crisis, myths, software engineering – a layered technology, software process, linear sequential model, prototyping model, RAD model, evolutionary software process model.

Unit **8Hrs**

Software Process and Project Metrics: Measures, metric indicators, metric in process and the project domains, software measurement, metrics for software quality.

Unit III **10Hrs**

Analysis Concepts and Principles: Requirement analysis, communication techniques, software prototyping & Specification.

Unit IV **10Hrs**

Analysis Modeling: Elements of the analysis model, data modeling, functional modeling, behavioral modeling, the mechanics of structured analysis, data dictionary, other classical analysis methods.

Unit-V **8Hrs**

Design Concepts and Principles: Software Design and software Engineering design process, Design principles, Design concepts, Design methods-Data design, Architectural design and process, Transform and Transaction mappings, Design post processing, Architectural design optimization, Interface design, Procedural design.

Unit VI **8Hrs**

Software Testing Methods: Fundamentals, Test case design, White box testing, basis path testing, control structure testing, black box testing, Software testing strategies.

Title of the Course with Code: Software Engineering CS-HC 4.2

After completion of this course students will be able to

CO	Statement
CO1	Analyze the process model chosen for the development of software and its merits and demerits.
CO2	Identify the clear, correct and consistent requirements for the project.
CO3	Design suitable data, architecture and user interface that copes with the requirements.

CO4	Estimate the cyclomatic complexity and design the corresponding test cases.
CO5	Conduct various integration testing approaches and note down pit falls in requirements, design and test cases.

References:

1. Roger S. Pressman, Software Engineering**,** 4/e, McGraw Hill.
2. I. Sommerville, Software Engineering, 6/e, Addison Wesley.
3. Shooman, Software Engineering, McGraw Hill .
4. T. C. Lethbridge and R. Laganiere, Object Oriented Software Engineering, Tat McGraw Hill.
5. Priestley, Practical Object Oriented Design using UML, TMH
6. Page Jones, Meiler, Fundamentals of object oriented design in UML.

CS-SC 4.3(a): Elective-I: Bioinformatics

Teaching: 4 hrs./week **Credits: 04**

Max Marks: 100 & Total Hours: 52

Cont. Assessment. 30

Code: **CS-SC** 4.3 (a) Univ Code :_____

Contact Hours : 4 hours per week Work load : 4 hours per week

Credit Points :04

Evaluation: Continuous Internal Assessment - 30 marks

Semester and Examination - 70 marks

UNIT-I **10Hrs**

Introduction- What is Bioinformatics, Goal, Scope, Applications, Limitations, and New Themes.

Basic Concepts of Molecular Biology - Life, Protein, Nucleic Acids, The Mechanism of Molecular Genetics, How the Genome Is Studied.

Unit II **10Hrs**

Introduction to Biological Databases- What is a Database? Types of Databases, Biological Databases, Pitfalls of Biological Databases, Information Retrieval from Biological Databases

Unit III **10Hrs**

Sequence Alignment: Pair wise Sequence Alignment, Database Similarity Searching, Multiple Sequence Alignment, Protein Motifs and Domain Prediction

Unit IV **10Hrs**

Gene Prediction: Categories of Gene Prediction Programs, Gene Prediction in Prokaryotes, Gene Prediction in Eukaryotes.

Unit V **8Hrs**

Molecular Phylogenetics: Phylogenetics Basics, Phylogenetic Tree Construction Methods and Programs.

Unit VI **4Hrs**

Genomics and Proteomics: Genome Mapping, Assembly, and Comparison and Proteomics

Title of the Course with Code: Bioinformatics CS-SC 4.3 (a)

After completion of this course students will be able to

CO	**Statement**
CO1	Knowledge and awareness of the basic principles bioinformatics and its applications
CO2	To get exposed to computational methods, tools and algorithms employed for Biological Data Interpretation.
CO3	Describe about the different types of Biological databases.
CO4	Overview about biological structure for prediction methods.

CO5	Understand the concepts of genomics and proteomics.

References

1. Xiong Jin, "*Essential Bioinformatics"*. Cambridge University Press, First South Asian edition.

2. Setubal Joao Carlos, Joao Meidanis, Jooao Carlos Setubal *"Introductionto ComputationalMolecularBiology",* Thomson Learning, First Reprint, 3003
3. Mount W David, *"BioinformaticsSequenceandGenomeAnalysis".* CBS Publishers, First Indian Reprint, 3005
4. Krane E Dan and Michael L Raymer, "*FundamentalConceptsofBioinformatics*". Pearson Education Inc., First Indian Reprint, 3003.

CS-SC 4.3 (b): Elective-I: Artificial Intelligence

Teaching: 4 hrs/week **Credits: 04**

Max Marks: 100 & Total Hours: 52

Code: **CS-SC** 4.3 (b) Univ Code :_____

Contact Hours: 4 hours per week Work load: 4 hours per week

Credit Points: 04

Evaluation: Continuous Internal Assessment -30 marks

Semester and Examination - 70 marks

Unit I **8Hrs**

General issues and overview of AI, AI Techniques, AI problems, AI Techniques, importance and areas of AI, problem solving state space search-DLF, BFS Production system, problem characteristics.

Unit II **8Hrs**

Heuristic Search Techniques: Generate and Test, Hill Climbing, Best First Search, Problem reduction, Constraint satisfaction- Cryptarithmetic and problems.

Unit III1 **10Hrs**

Knowledge representation & mapping, approaches to knowledge to representation, issues in knowledge representation, Representing simple facts in logic, representing instance and relationships,

Resolution and natural deduction Representing knowledge using rules, Procedural v/s Declarative

knowledge, Logic programming, Forward v/s Backward chaining, Matching & control knowledge.

Unit IV **10Hrs**

AI programming language: Prolog- objects, relationships, facts, rules and variables, Prolog: Syntax and data structures, representing objects & relationships by using "trees" and "lists", use of cut, I/O of characters and structures.

Unit V **10Hrs**

Symbolic reasoning under uncertainty: Introduction to monotonic reasoning, Logics for Nonmonotonic reasoning, implementation issues, implementation: DFS & BFS.

Unit VI **6Hrs**

Slot and filler structures: Semantic nets, frames, conceptual dependency, scripts.

Title of the Course with Code: Artificial Intelligence CS-SC 4.3 (b)

After completion of this course students will be able to

CO	Statement
CO1	Apply the knowledge of Artificial Intelligence to write simple algorithm for agents.
CO2	Apply AI knowledge to solve problem on search algorithm.
CO3	Develop knowledge base sentences propositional logic and first order logic

CO4	Apply first order logic to solve knowledge engineering process
CO5	Apply and analyze the knowledge of machine learning

References:

1. Rich & Knight , Artificial Intelligence, TMH
2. Cloksin & Mellish , Programming In Prolog, Narosa Publishing House.
3. Nillson Harcourt, Principles of Artificial Intelligence, Asia & Morgan.

Janakiraman, Sarukesi & Gopal Krishnan Macmillan. Foundation Of Artificial Intelligence & Expert System, MacMillan.

CS-SC 4.3(c): Elective-I: Neural Networks and Fuzzy Systems

Teaching: 4 hrs. /week **Credits: 04**

Max Marks: 100 & Total Hours: 52 Cont.Assessments.30

Code: **CS-SC** 4.3(c) Univ Code: _____

Contact Hours: 4 hours per week Work load: 4 hours per week

Credit Points: 04

Evaluation: Continuous Internal Assessment -30 marks

Semester and Examination - 70 marks

Unit I **10Hrs**

Introduction: Introduction to Neural networks and fuzzy logic, basic concepts of neural networks, human brain, model of artificial neuron, neural network architectures, Characteristics of neural networks, learning methods.

Unit–II **12Hrs**

Backpropogation Networks: Architecture, backpropogation learning, applications, tuning of backpropogation neural networks, parameters in BPN, variation of standard backpropogation algorithm, research directions.

Unit III **8Hrs**

Associative Memory: Autocorrelators, heterocorrelators, Wnag etc. al.'s multiple training encoding strategy, exponential BAM, associative memory for real-coded patter pairs, applications.

Unit IV **8Hrs**

Adaptive Resonance Theory: Classical ART networks, simplified ART architecture, ART1, ART2, applications.

Unit V **8Hrs**

Fuzzy Set Theory: Crisp sets, Fuzzy sets, Crisp relations, Fuzzy relations.

Unit VI **6Hrs**

Fuzzy Systems: Crisp logic, predicate logic, fuzzy logic, fuzzy rule based systems, defuzzification methods, and applications.

Title of the Course with Code: Neural Networks and Fuzzy Systems CS-SC 4.3(c)

After completion of this course students will be able to

CO	Statement
CO1	Describe and analysis of neural network architectures.
CO2	Design neural network approach to a particular problem.
CO3	Understand the concepts of Fuzzy systems and its applications.

References:

1. S. Rajashekaran, G. A. Vijayalakshmi Pai, Neural Networks, Fuzzy Logics and

Genetic Algorithms, PHI.

2. Stamatios V. Kartalopoulos, Understanding Neural Networks And Fuzzy Logic—Basic Concepts And Applications, PHI (30056.
3. Bart Kosko, Neural networks and fuzzy systems - A dynamical systems approach tomachineintelligencePHI.

CS-SC 4.3 (d): Elective-I: Theory of Computation

Teaching: 4 hrs./week **Credits: 04**

Max Marks: 100 & Total Hours: 52

Code : **CS-SC** 4.3 (d) Univ Code :_____

Contact Hours : 4 hours per week Work load : 4 hours per week

Credit Points :04

Evaluation: Continuous Internal Assessment -30 marks

Semester and Examination -70 marks

Unit–I **10Hrs**

Introduction: Sets, relations and functions; strings and their properties; automation, transition systems, nondeterministic finite state machines, equivalence of DFA and NDFA, Mealy and Moore Models.

Unit II **14Hrs**

Formal Languages and Regular Grammars: Chomsky classification of languages, languages and their relation, operations on languages, languages and automata, regular expressions, finite automata and regular expressions, pumping lemma, regular sets and regular grammars.

Unit III **14Hrs**

Context-free languages: Context-free languages and derivation trees, ambiguity in context-free grammars, normal forms for context-free grammars, decision algorithms, push down automata, pushdown automata and context-free languages, parsing and pushdown automata.

Unit IV **14Hrs**

Turing Machines and Linear Bounded Automata: Turing machine model, representation of Turing machines, language acceptability, design of Turing machines, the model of linear bounded automation, Turing machines and type 0 grammars, linear bounded automata and languages, halting problem of completeness, NP-completeness.

Title of the Course with Code: Theory of Computation CS-SC 4.3 (d)

After completion of this course students will be able to

CO	**Statement**
CO1	Understand formal machines, languages
CO2	Design finite state machines for acceptance of strings.
CO3	Design context free grammars for formal languages
CO4	Develop pushdown automata acceptance strings
CO5	Design Turing machine.

References:

1. K.L.P. Mishra and N. Chandrasekaran, Theory of Computer Science, 2/e, PHI.
2. Michael Sipser, Introduction to the Theory of Computation, Thomson Learning.
3. J P Hoperoft, J D Ullman, Introduction to Automata, Languages and Computation, Narosa Publications.
4. John C. Martin, Introduction to Languages and the Theory of Computation,2nd Edition, McGraw Hill.

CS-SC 4.3 (e): Elective-I: Pattern Recognition

Teaching: 4 hrs./week **Credits: 04**

Max Marks: 100 & Total Hours: 52

Code: **CS-SC** 4.3(e) Univ Code :_____

Contact Hours: 4 hours per week Work load: 4 hours per week

Credit Points: 04

Evaluation: Continuous Internal Assessment -30 marks

Semester and Examination -70 marks

Unit I **6Hrs**

Introduction: Application of Pattern Recognition, statistical Decision Theory, Image Processing and Analysis.

Unit II 8Hrs Probability: Introduction, Probability of Events, Random Variables, Joint Distribution and Densities, Moments of Random variables, Estimation of Parameters from samples, Minimum Risk Estimations.

Unit III **12Hrs**

Statistical Decision Making: Introduction, Baye's Theorem, Multiple Features, Conditionally Independent Features, Decision Boundaries,- Estimation of Error rates, Characteristic centers, Estimating the Composition of Populations.

Unit IV 10Hrs Non Parametric Decision Making: Introduction, Histograms, Kernel and Windows Estimators, Nearest Neighbour Classification Techniques, Adaptive Decision Boundaries, Adaptive Discriminant Functions, Minimum Squared.

Unit V **8Hrs**

Clustering: Introduction, Hierarchical Clustering, Partitional Clustering.

Unit VI **10Hrs**

Artificial Neural Networks: Introduction, Nets without Hidden layers, Nets with Hidden layers,The Back – Propagation Algorithm, Hopfied Nets – An Application: Classifying Sex from facial images.

Title of the Course with Code: Pattern Recognition CS-SC 4.3(e)

After completion of this course students will be able to

CO	Statement
CO1	Defining the concepts of pattern recognition
CO2	Distinguish procedures, methods and algorithms related to pattern recognition.
CO3	Design and develop a pattern recognition system for the specific application.
CO4	Able to apply various decision making and clustering algorithms in pattern recognition systems

References:

1. Earl Gose, Richard Johnsonbaugh and Steve Jost, Pattern Recognition and Image Analysis, PHI, 1997.
2. Fu.K.S., Syntactic Methods in Pattern Recognition, Academic Press, 1974.
3. Tray Y Young and Thomas W Calvert, Classification, Estimation and Pattern Recognition, American Elservier Publication Company Inc., 1994.
4. Duda R.O. and Hart P.E., Pattern Classification and Scene Analysis, John Wiley.

CS-SC 4.4(a): Elective-II: Data Warehousing and Mining

Teaching: 4 hrs./week **Credits: 04**

Max Marks: 100 & Total Hours: 52 Cont.

Assessment. 30

Code : **CS-SC** 4.4 (a) Univ Code :

Contact Hours : 4 hours per week Work load : 4 hours per week

Credit Points :04

Evaluation: Continuous Internal Assessment - 30marks

Semester and Examination - 70marks

UNIT-I **12Hrs**

Introduction: Fundamentals of data mining, Data Mining Functionalities, Classification of Data Mining systems, Major issues in Data Mining, Data Warehouse and OLAP Technology for Data Mining Data Warehouse, Multidimensional Data Model, Data Warehouse Architecture, Data Warehouse Implementation, Further Development of Data Cube Technology, From Data Warehousing to Data Mining,

UNIT-II **12Hrs**

Data Preprocessing: Needs Preprocessing the Data, Data Cleaning, Data Integration and Transformation, Data Reduction, Discretization and Concept Hierarchy Generation, Online Data Storage.

Data Mining Primitives, Languages, and System Architectures: Data Mining Primitives, Data Mining Query Languages, Designing Graphical User Interfaces Based on a Data Mining Query Language Architectures of Data Mining Systems

UNIT-III **10Hrs**

Concepts Description: Characterization and Comparison: Data Generalization and Summarization-Based Characterization, Analytical Characterization: Analysis of Attribute Relevance, Mining Class Comparisons: Discriminating between Different Classes, Mining Descriptive Statistical Measures in Large Databases.

UNIT-IV **10Hrs**

Mining Association Rules in Large Databases: Association Rule Mining, Mining Single-Dimensional Boolean Association Rules from Transactional Databases, Mining Multilevel Association Rules from Transaction Databases, Mining Multidimensional Association Rules from Relational Databases and Data Warehouses, From Association Mining to Correlation Analysis, Constraint-Based Association Mining.

UNIT-V **08 Hrs**

Classification and Prediction: Issues Regarding Classification and Prediction, Classification by Decision Tree Induction, Bayesian Classification, Classification by Back propagation, Classification Based on Concepts from Association Rule Mining, Other Classification Methods.

Title of the Course with Code: Data Warehousing and Mining CS-SC 4.4 (a)

After completion of this course students will be able to

CO	**Statement**
CO1	Understanding the fundamental concepts of data mining and its stages
CO2	Apply preprocessing techniques on the data.
CO3	Understand and apply association rules for large databases.
CO4	Analyze classification and prediction algorithms.

References:

1. Jiawei Han & Micheline Kamber , Data Mining – Concepts and Techniques, Harcourt India.
2. Arun K Pujari , Data Mining Techniques , University Press
3. W. H. Inmon,, Building the DataWarehouse-, Wiley Dreamtech India Pvt. Ltd..
4. Sam Anahory & Dennis Murray, Data Warehousing in the Real World, Pearson Edn Asia.
5. Paulraj Ponnaiah , Data Warehousing Fundamentals, Wiley Student Edition
6. Ralph Kimball , The Data Warehouse Life cycle Tool kit –Wiley Student Edition
7. Margaret H Dunham , Data Mining Introductory and advanced topics, Pearson Education .

CS-SC 4.4 (b): Elective-II: Embedded Systems

Teaching: 4 hrs./week **Credits: 04**

Max Marks: 100 & Total Hours: 52

Code : **CS-SC** 4.4 (b) Univ Code :_____

Contact Hours : 4 hours per week Work load : 4 hours per week

Credit Points :04

Evaluation: Continuous Internal Assessment -30 marks

Semester and Examination -70 marks

Unit I **8Hrs**

An overview of embedded systems: Introduction to embedded systems, Categories and requirements of embedded systems, Challenges and issues related to embedded software development, Hardware/Software co-design, Introduction to IC technology, Introduction to design technology.

Unit II **12Hrs**

Embedded Software development: Concepts of concurrency, processes, threads, mutual exclusion and inter-process communication, Models and languages for embedded software, Synchronous approach to embedded system design, Scheduling paradigms, Scheduling algorithms, Introduction to RTOS, Basic design using RTOS

Unit III **10Hrs**

Embedded C Language: Real time methods, Mixing C and Assembly, Standard I/O functions, Preprocessor directives, Study of C compilers and IDE, Programming the target device

Unit IV **12Hrs**

Hardware for embedded systems: Various interface standards, Various methods of interfacing, Parallel I/O interface, Blind counting synchronization and Gadfly Busy waiting, Parallel port interfacing with switches, keypads and display units, Memory and high speed interfacing, Interfacing of data acquisition systems, Interfacing of controllers, Serial communication interface, Implementation of above concepts using C language.

Unit V **10Hrs**

Case studies and Applications of embedded systems: Applications to: Communication, Networking, Database, Process Control, Case Studies of: Digital Camera, Network Router, RTLinux.

Title of the Course with Code: Embedded Systems CS-SC 4.4 (b)

After completion of this course students will be able to

CO	Statement
CO1	Able to demonstrate knowledge of basic principles of embedded systems
CO2	Understand the various important parameters required for the development of embedded software.
CO3	Understand the software and hardware requirements for the development of embedded systems.
CO4	Study on case studies and applications of embedded systems.

References:

1. Raj Kamal, Embedded Systems, Tata McGraw Hill.
2. David E. Simon, An Embedded Software Primer, Pearson Education.
3. Muhammad Ali Mazidi and Janice Gillispie Mazidi, The 7051Microcontroller and Embedded Systems, Pearson Education.
4. Frank Vahid, Tony Givargis, Embedded System Design: A Unified Hardware/Software Introduction, John Wiley.
5. Craig Hollabaugh, Embedded Linux, Pearson Education
6. Daniel Lewis, Fundamentals of Embedded Software, Pearson Education.
7. Barnett, Cox, O'Cull, Embedded C Programming and the Atmel AVR , Thomson Learning
8. Myke Predko, Programming and Customizing the 7051 Microcontroller, TMH

CS-SC 4.4 (c): Elective-II: Advanced Computer Architecture

Teaching: 4 hrs./week **Credits: 04**

Max Marks: 100 & Total Hours: 52

Code : **CS-SC** 4.4 (c) Univ Code :_____

Contact Hours : 4 hours per week Work load : 4 hours per week

Credit Points :04

Evaluation: Continuous Internal Assessment -30 marks

Semester and Examination -70 marks

Unit I **10Hrs**

Pipe Line And Vector Processing: Introduction , Linear pipepline , Classification, Reservation tables, Introduction prefetch and branch handling, Data Buffering and Busing structure, Internal forwarding and register tagging, Hazard detection , Characteristics of Vector processing.

Unit II **14Hrs**

Array Processing: SIMD Array processors, SIMD Interconnection networks , Static and dynamic - Mesh connection, Cube connection, Barrel shifter and data manipulation, parallel algorithm for SIMD matrix multiplication.

Unit III **14Hrs**

Multiprocessor Architecture: Loosely coupled, tightly coupled multiprocessor configurations, Interconnection networks, Interleaved memory organization, Multiprocessor operating systems, Software requirements for multiprocessors.

Unit IV **14Hrs**

Multiprocessing Control and Algorithms: Inter process communication mechanism and process synchronization, system deadlock problem, Multiprocessor scheduling strategy, parallel algorithms for multiprocessors.

Title of the Course with Code: Advanced Computer Architecture CS-SC 4.4 (c)

After completion of this course students will be able to

CO	**Statement**
CO1	Understand the concepts of computer architecture
CO2	Understand the concept of Array Processing, Parallel algorithms for SIMD
CO3	Analyze various multiprocessor architecture and its software requirements for multiprocessors.

References:

1. Kai Hwang and Feye A. Briggs, Computer Architecture and parallel processing, McGraw Hill.
2. Dezso Sima, Terence Fountain and Peter Kacsuk, Advanced Computer Architecture-A Design Space Approach, Pearson Education (3005)
3. Kain, Advanced Computer Architecture-A Systems Design Approach, PHI(3006).
4. Kai Hwang, Advanced Computer Architecture, McGraw Hill (3000).

CS-SC 4.4 (d): Elective-II: Mobile Communications

Teaching: 4 hrs./week **Credits: 04**

Max Marks: 100 & Total Hours: 52

Code : **CS-SC** 4.4 (d) Univ Code :

Contact Hours : 4 hours per week Work load : 4 hours per week

Credit Points :04

Evaluation: Continuous Internal Assessment - 30 marks

Semester and Examination - 70 marks

Unit I **12Hrs**

Introduction: History of wireless communication, a simplified reference model, applications, frequencies for radio transmission, signals, antennas, signal propagation, multiplexing, modulation, spread spectrum, cellular systems.

Unit II **8Hrs**

Medium access control: SDMA, FDMA, TDMA, CDMA

Unit III **8Hrs**

Telecommunications ans satellite systems: GSM, DELT, TETRA, UMTS, and IMT-3000, basics of satellite systems, routing.

Unit IV **8Hrs**

Broadcast Systems: Cyclical repition of data, digital audio broadcasting, digital video broadcasting.

Unit V **8Hrs**

Wireless Lan: infrared vs radio transmission, infrastructure and adhoc network, Blue Tooth.

Unit VI **8Hrs**

Mobile Network Layer and Transport Layer: Mobile IP, dynamic host configurartion protocol, mobile adhoc networks, traditional TCP, classical TCP improvements.

Title of the Course with Code: Mobile Communications CS-SC 4.4 (d)

After completion of this course students will be able to

CO	**Statement**
CO1	To make students familiar with various generations of mobile communications
CO2	To understand the concept of cellular communication
CO3	To understand the basics of wireless communication

CO4	Knowledge of GSM mobile communication standard, its architecture, logical channels, advantages and limitations.
CO5	Knowledge of 4G mobile standards and their comparison with 3G technologies

References:

1. Johen Schiller, Mobile Communications, 2/e, Pearson Education.
2. Stuber G.L., Principles of Mobile Communications, Academic Press.
3. Rappapert T.S., Wireless Communication Principles & Practices, Prentice Hall.

CS-SC 4.4(e): Elective-I: Digital Image Processing

Teaching: 4 hrs/week **Credits: 04 Max Marks: 100 & Total Hours: 52**

Code: **CS-SC** 4.4 (e) Univ Code: _____

Contact Hours: 4 hours per week Work load: 4 hours per week

Credit Points: 04

Evaluation: Continuous Internal Assessment -30 marks

Semester and Examination -70 marks

Unit I **8Hrs**

Digital Image Fundamentals: The origins of Digital Image Processing, Examples of Fields that Use Digital Image Processing, Fundamentals Steps in Image Processing, Elements of Digital Image Processing Systems, Image Sampling and Quantization, Some basic relationships like Neighbours, Connectivity, Distance Measures between pixels, Linear and Non Linear Operations.

Unit II **8Hrs**

Image Enhancement in the Spatial Domain: Some basic Gray Level Transformations, Histogram Processing, Enhancement Using Arithmetic and Logic operations, Basics of Spatial Filters, Smoothening and Sharpening Spatial Filters, Combining Spatial Enhancement Methods.

Unit III **10Hrs**

Image Enhancement in the Frequency Domain: Introduction to Fourier Transform and the frequency Domain, Smoothing and Sharpening Frequency Domain Filters, Homomorphic Filtering.

Unit IV **10Hrs**

Image Restoration: A model of The Image Degradation / Restoration Process, Noise Models, Restoration in the presence of Noise Only Spatial Filtering, Periodic Noise Reduction by Frequency Domain Filtering, Linear Position-Invariant Degradations, Estimation of Degradation Function, Inverse filtering, Wiener filtering, Constrained Least Square Filtering, Geometric Mean Filter, Geometric Transformations.

Unit V **10Hrs**

Image Compression: Coding, Interpixel and Psychovisual Redundancy, Image Compression models, Elements of Information Theory, Error free comparison, Lossy compression, Image compression standards.

Unit VI **08 Hrs**

Image Segmentation: Detection of Discontinuities, Edge linking and boundary detection, Thresholding, Region Oriented Segmentation, Motion based segmentation.

Title of the Course with Code: Digital Image Processing CS-SC 4.4 (e)

After completion of this course students will be able to

CO	**Statement**
CO1	Review the fundamental concepts of digital image processing system.
CO2	Analyze images in the frequency domain using various transforms.
CO3	Analyze the concepts of image compression and segmentation.

References:

1. R.C. Gonzalez and R. E. Words, Digital Image Processing, 2/e, Pearson Education.
2. Anil K .Jain, Fundamentals of Digital Image Processing, PHI.
3. W.K. Pratt, Digital Image Processing,Wiley Eastern.
4. Chanda & Mujumder, Digital Image Processing and Analysis, PHI.
5. Millan Sonka, Vaclav Hlavac and Roger Boyle, Image Processing Analysis and Machine Vision, Thomson Learning-Vikas Publishing House .
6. Joshi, Digital Image Processing – An algorithmic approach, PHI.

Chairman
CHAIRMAN
Department of Computer Science
Vijayanagar Sri Krishnadevaraya University
Jnana Sagar Campus, Vinayaknagar
Cantonment, BALLARI-583104