Tentative Scheme and Syllabi

Tentative Scheme and Syllabi 1
Tentative UG(AI&DE) Scheme 2
Department of Computer Science and Engineering 2
Semester-wise Scheme and Syllabus 7
Scheme and Syllabus of 1st Year Institute Core Subjects 7
Scheme and Syllabus of 1st Semester 9
Scheme and Syllabus of 2nd Semester 13
Scheme and Syllabus of 3rd Semester 17
Scheme and Syllabus of 4th Semester 30
Scheme and Syllabus of 5th Semester 41
Scheme and Syllabus of 6th Semester 53

Tentative UG(AI&DE) Scheme

Department of Computer Science and Engineering

First Semester					
S. No	**Code**	**Subject**	**L-T-P**	**Credit**	**Type**
1		***Institute Core Subjects***		***19***	***IC***
2	22AIT1xx	Discrete Mathematics	3-0-0	3	PC
3	22AIT1xx	Problem Solving using C	2-0-0	2	PC
4	22AIP1xx	Problem Solving Using C Lab	0-0-2	1	PC
				25	

Second Semester					
S. No	**Code**	**Subject**	**L-T-P**	**Credit**	**Type**
		Institute Core Subjects		***18***	***IC***
	22AIT1xx	Data Structures	3-0-0	3	PC
	22AIT1xx	Mathematics for AI	3-0-0	3	PC
	22AIP1xx	Data Structures Lab	0-0-2	1	PC
				25	

Third Semester					
S. No	**Code**	**Subject**	**L-T-P**	**Credits**	**Type**
	22AIT2xx	Digital Systems and Computer Architecture	3-1-0	4	PC
	22AIT2xx	Design and Analysis of Algorithms	3-0-0	3	PC
	22AIT2xx	Operating Systems	3-0-0	3	PC
	22AIT2xx	Foundations of Data Science	3-0-0	3	PC
	22AIT2xx	Theory of Computation	3-1-0	4	PC
	22HST2xx	Social Sciences and Professional Ethics	3-1-0	4	BS
	22AIP2xx	Data Science Lab	0-0-2	1	PC
	22AIP2xx	Design and Analysis of Algorithms Lab	0-0-4	2	PC
	22AIP2xx	Operating Systems Lab	0-0-2	1	PC
			29	**25**	

Fourth Semester					
S. No	**Code**	**Subject**	**L-T-P**	**Credits**	**Type**
	22AIT2xx	Artificial Neural Networks	3-0-0	3	PC
	22AIT2xx	Artificial Intelligence	3-0-0	3	PC
	22AIT2xx	Compiler Design	3-0-0	3	PC
	22AIT3xx	Computer Networks	3-0-0	3	PC
	22AIT2xx	Database Management Systems	3-0-0	3	PC
	22MMTxx	Basics of Management	3-0-0	3	PLEAS
	22AIP3xx	Computer Networks Lab	0-0-4	2	PC
	22AIP2xx	Artificial Intelligence Lab	0-0-4	2	PC
	22AIP2xx	Database Management Systems Lab	0-0-2	1	PC
	22AIP2xx	Open-ended Minor Project	0-0-4	2	PC
			32	**25**	

Fifth Semester					
S. No	**Code**	**Subject**	**L-T-P**	**Credits**	**Type**
	22AIT3xx	Digital Image Processing	3-0-0	3	PC
	22AIT2xx	Machine Learning	3-0-0	3	PC
	22AIT3xx	Big Data Analytics	3-0-0	3	PC
	22AIT2xx	Information Retrieval	3-0-0	3	PC
	22AIT3xx	Data Mining and Warehousing	3-0-0	3	PC
	22AIT3xx	Program Elective-1	3-0-0	3	PE
	22AIP3xx	Digital Image Processing Lab	0-0-2	1	PC
	22AIP2xx	Machine Learning Lab	0-0-4	2	PC
	22AIP3xx	Data Analytics lab	0-0-2	1	PC
			26	**22**	

Honors					
	22AITxxx	Bio Medical Image Analysis		3	
	22AITxxx	Social Network Analysis		3	
				6	

Minor AIDE					
	22AIT1xx	Data Structures	3-0-0	3	PC
	22AIT2xx	Foundations of Data Science	3-0-0	3	PC
				6	

Sixth Semester

S. No	Code	Subject	L-T-P	Credits	Type
	22AIT3xx	Deep Learning	3-0-0	3	PC
	22AIT3xx	Natural Language Processing	3-0-0	3	PC
	22AIT3xx	High Performance Computing	3-0-0	3	PC
	22AIT3xx	Information Security	3-0-0	3	PC
	22AIT3xx	Program Elective-2	3-0-0	3	PE
	22ECxxx	Wireless and 5G Communication	3-0-0	3	PLEAS
	22AIP3xx	Deep Learning Lab	0-0-4	2	PC
	22AIP3xx	Natural Language Processing Lab	0-0-2	1	PC
	22AIP3xx	High Performance Computing Lab	0-0-2	1	PC
			27	**22**	

Honors					
	22AITxxx	Honors Elective-1		3	
	22AITxxx	Honors Elective-2		3	
				6	

Minor AIDE					
	22AIT2xx	Database Management Systems	3-0-0	3	PC
	22AIT2xx	Artificial Intelligence	3-0-0	3	PC
				6	

Seventh Semester					
S. No	**Code**	**Subject**	**L-T-P**	**Credits**	**Type**
1	22AIS401	Training Seminar	0-0-4	2	PC
2	22AID402	Minor Project	0-0-6	3	PC
3	22AITxxx	Program Elective-3	3-0-0	3	PE
4	22AITxxx	Program Elective-4	3-0-0	3	PE
5	22AITxxx	Program Elective-5	3-0-0	3	PE
6	22AITxxx	Program Elective-5 Lab	0-0-2	1	PE
7		Open Elective – 1	3-0-0	3	OE
			24	**18**	

Honors					
	22AITxxx	Honors Elective-3*		3	
				3	

Minor AIDE					
	22AIT3xx	Big Data Analytics	3-0-0	3	PC
				3	

Eighth Semester					
S. No	**Code**	**Subject**	**L-T-P**	**Credits**	**Type**
1	22AID403	Major Project	0-0-16	8	PC
2	22AITxxx	Program Elective-6	3-0-0	3	PE
3	22AITxxx	Program Elective-7	3-0-0	3	PE
4	22AIPxxx	Program Elective-7 Lab	0-0-2	1	PE
5		Open Elective – 2	3-0-0	3	OE
			27	**18**	

Honors					
	22AITxxx	Honors Elective-4*		3	
				3	

Minor AIDE					
	22AIT3xx	Deep Learning	3-0-0	3	PC
				3	

** Honors Elective courses will be taken from PG departmental subject pool*

 # For Program Elective 5 & 7, only elective with Lab components to be selected

Programming with Python					
Prerequisite: Nil		L	T	P	C
Total hours: 28		2	0	0	2
Course Content					Hrs.
Unit 1	Introduction to computer system and binary number systems – addition, subtraction (2's complement), multiplication, left shifting and right shifting.				4
Unit 2	Introduction to Python: Python variables, Python basic Operators, Understanding python blocks. Python Data Types, Declaring and using Numeric data types: int, float etc. Python Program Flow Control Conditional blocks: if, else and else if, Simple for loops in python, for loop using ranges, string, list and dictionaries. Use of while loops in python, Loop manipulation using pass, continue, break and else. Programming using Python conditional and loop blocks.				6
Unit 3	Python Complex data types: Using string data type and string operations, Defining list and list slicing, Use of Tuple data type. String, List and Dictionary.				6
Unit 4	Building blocks of python programs: string manipulation methods, List manipulation, Dictionary manipulation, Programming using string, list and dictionary in-built functions. Python Functions, organizing python codes using functions, Introduction to classes.				6
Unit 5	Python File Operations: Reading files, writing files in python, Case study: development of mini projects using libraries, such as, pandas, matplotlib, NumPy, SciKit-learn, Seaborn, etc.				6

References

1.	Wesley J. Chun, "Core Python Applications Programming", 3rd Edition, Pearson Education, 2016.
2.	Charles Dierbach, "Introduction to Computer Science using Python", Wiley, 2015.
3.	Jeeva Jose & P. Sojan Lal, "Introduction to Computing and Problem Solving with PYTHON", Khanna Publishers, New Delhi, 2016.
4.	Downey, A. et al., "How to think like a Computer Scientist: Learning with Python", John Wiley, 2015.
5.	Mark Lutz, "Learning Python", 5th edition, O'reilly Publication, 2013, ISBN 978- 1449355739
6.	John Zelle, "Python Programming: An Introduction to Computer Science", Second edition, Course Technology Cengage Learning Publications, 2013, ISBN 978- 1590282410
7.	Michel Dawson, "Python Programming for Absolute Beginners", Third Edition, Course Technology Cengage Learning Publications, 2013, ISBN 978-1435455009
8.	David Beazley, Brian Jones., "Python Cookbook", Third Edition, O'reilly Publication, 2013, ISBN 978- 1449340377

Programming with Python Lab				
Prerequisite:	L	T	P	C
Total hours: 28	0	0	0	2

	Course Content	Hrs.
	The following proposed coverage are broad guiding areas lab. The programs mentioned here just sample programs and they are just for reference purpose. The instructor offering the course in consultation with the theory offered can adopt further variations in tune with concerned theory course. 1. Installation of Python Tool, Introduction to Python programming, and python datatypes [1 Lab] 2. Data types, Input/Output and library imports [1 Lab] 3. Python strings operations, Doc strings [1 Lab] 4. Objects - List, Tuples and Dictionaries [3 Lab] 5. Control flow, functions working and some advanced functions [2 Lab] 6. Python File Operations: Reading files, Writing files in python [1 Lab] 7. Introduction to classes [1 Lab] 8. NumPy, Matlabplotlib utility functions [2 Lab]	

References:	
1.	Core Python Applications Programming: Wesley J. Chun, Pearson Education, 2016.
2.	Introduction to Computer Science using Python: Charles Dierbach, Wiley, 2015
3.	Python for Programmers: Paul J. Deitel, Harvey Deitel , Pearson, 2020 .
4.	Learning Python: Mark Lutz, Orelly Publication, 2013
5.	Python Programming: An Introduction to Computer Science: John Zelle, Course Technology Cengage Learning Publications, 2013.

Scheme and Syllabus of 1st Semester

First Semester					
S. No	**Code**	**Subject**	**L-T-P**	**Credit**	**Type**
		Institute Core Subjects		***19***	***IC***
	AIT1xx	Discrete Mathematics	3-0-0	3	PC
	AIT1xx	Problem Solving using C	2-0-0	2	PC
	AIP1xx	Problem Solving Using C Lab	0-0-2	1	PC
				25	

Discrete Mathematics					
Prerequisite: Nil		L	T	P	C
Total hours: 42		3	0	0	3
Course Content					Hrs.
Unit 1	Logic: Truth Tables, Conditionals (P ⇒ Q), and Bi-conditionals (P ⇔ Q), Negation, Converse, and Contrapositive, Existential and Universal Quantifiers (∀,∃, ∃!), Proof Techniques (Contrapositive, Contradiction, Induction), Counterexamples, and Proving Statements with Quantifiers, Predicate logic, first order logic, Logical Inferences.				8
Unit 2	Set Theory: Sets and Set Notation, the Empty Set, the Power Set, Cardinality rules and infinite sets, Union, Intersection, Complement, Subsets , Proving sets are equal, Axioms of Naïve Set Theory.				6
Unit 3	Relations: Cartesian Products and Relations, Equivalence Relations and Partitions, Partial Orderings, Lattices.				6
Unit 4	Functions: Definition of a Function, Domains and Co-domains, Composition and Inverses, Well-Defined, Injective, Surjective, and Bijective Functions, Recurrence Relations, Generating functions.				6
Unit 5	Abstract Algebra: Groups-Binary operation, and its properties, Definition of a group, Groups as symmetries, cyclic, dihedral, symmetric, matrix groups, Subgroups, Cosets, normal subgroups and quotient groups, Conjugacy classes, Lagrange's theorem, Monoid.				8
Unit 6	Number Theory: Prime Numbers, Euclid's Algorithm for GCD, The GCD- LCM product theorem, Extended Euclid's Algorithm, Linear Diophantine Equations, Modular Arithmetic, Chinese Reminder Theorem, Fast Modular Exponentiation, Fermat's little theorem, Euler's totient theorem, Euler's theorem.				8

References	
1.	Ronald L. Graham, Donald E. Knuth, Oren Patashnik ,Concrete Mathematics: A Foundation for Computer Science (2nd Edition)
2.	K. Rosen, Discrete Mathematics and Its Applications, 7th edition, McGraw-Hill, 2011.
3.	M. Lipson, Schaum's Outline of Discrete Mathematics, revised 3rd edition, 2009.
4.	D. Velleman, How to Prove it: A Structured Approach. Cambridge University Press, 1994

Problem solving using C				
Prerequisite: :Nil	L	T	P	C
Total hours: 28	2	0	0	2

	Course Content	Hrs
Unit 1	Introduction to Computers, Basic Computer Organization, Computational Thinking and problem solving, Planning the Computer Program - Debugging, Types of errors, Techniques of Problem. Aspects of programming language: Syntax, semantics. System Software, Application Software. Compiler -Compilation process - Compiler and interpreter. Basics: C language introduction, C language Standards, Data Types and Storage Classes: Different data types, Storage Classes – auto, static, extern, register. Reserved words, operators, constants in C, identifiers, printf/scanf (formatted printf/scanf), assignment statement, built-in data types – int, char, float, double; usage of sizeof(), integer arithmetic, typecasting	6
Unit 2	IF/IF..ELSE control construct through maximum of two numbers, ternary operator for maximum of three numbers. SWITCH statement through figure to words problem Swapping of variables, Solving problem of gcd of two numbers. Introduction to 1D arrays in C, implementation of strings as char array, string function implementation: example problem could be palindrome. Loop constructs: significance of initialization, terminating condition and increment/decrement (pre/post increment/decrement operator usage). Usage of FOR/WHILE/DO..WHILE in problems like sum /maximum/ deviation of N numbers. Illustration of loops for solving computation of sin of a number	8
Unit 3	Pointers: Introduction to pointers, pointer arithmetic, void *, pointers v/s array, malloc() – case study linked list. Pointer to array versus array of pointers, pointers to structures, array of pointers, Pointer to functions. Enum operator. File Handling in C: Basics of working with text files, File read, write, append and other similar operations.	
Unit 4	Problem Solving: Sorting an array consisting of zeros and ones, Partitioning an array, merging two sorted arrays, computation of square root of a number Recurrence through Factorial problem, binary search to illustrate divide and conquer approach, Fibonacci through recursion and problems with this approach, Fibonacci through storing previous values – introduction to dynamic programming, Nested loops through sorting methods; use of break and continue Bit vector implementation of set and usage of bitwise operators for testing membership (withing set), union and intersection of two sets. Macro & Preprocessor in C	6
Unit 5	Structures in C: struct and typedef through implementation of complex numbers Functions: Passing arguments in main() function, Call by value, Call by reference. Function for implementing raising a number to large power (logarithmic complexity). Multi-dimensional array (example problem can be matrix transpose/ addition) Command line arguments in C Passing variable number of arguments	8

References

1.	Education Solutions Limited, I. T. L. (2004). Introduction to Computer Science. India: Pearson Education.
2.	How to Solve it by Computer, RG Dromey, PHI
3.	The C Programming Language, Brian W. Kernighan and Dennis Ritchie, Latest Edition, Prentice Hall.
4.	Programming in ANSI C, E. Balagurusamy, Latest Edition, McGraw Hill
5.	Let us C, YashavantKanetkar, Latest Edition, BPB Publication

<table>
<tr><th colspan="6">Problem solving using C Lab</th></tr>
<tr><td colspan="2">Prerequisite:</td><td>L</td><td>T</td><td>P</td><td>C</td></tr>
<tr><td colspan="2">Total hours: 28</td><td>0</td><td>0</td><td>2</td><td>1</td></tr>
<tr><th colspan="5">Course Content</th><td>Hrs.</td></tr>
<tr><td>Unit 1</td><td colspan="4">The following proposed coverage are broad guiding areas lab. The programs mentioned here just sample programs and they are just for reference purpose. The instructor offering the course in consultation with the theory offered can adopt further variations in tune with concerned theory course.
1. Basic C commands and First C program-printing hello world on the screen, programs related to basic arithmetic operations, swapping of numbers etc. (2 lab)
2. C Expressions: Programs involving concepts of C expressions like finding roots of quadratic equation, area of circle and simple interest calculation. (1 lab)
3. C operators: Programs requiring in-depth knowledge of various C operators (especially conditional operator, bitwise operators and sizeof operator). (1 lab)
4. Conditional statements: Programs with applications of c conditional statements: if, if else, nested if else, switch-case (1 lab)
5. Arrays and Loops: C programs for performing various operations (finding maximum, second-maximum, minimum, reversing an array etc) on 1-D arrays and Applications of concepts of loops (leap year, palindrome, displaying prime numbers etc). (2 lab)
6. Functions and Recursions: Programs demonstrating use of functions (like adding N numbers, calculator etc) and Recursion (factorial, Fibonacci, GCD, binary search etc). (1 lab)
7. Strings, Pointers and Structures: Programs related to the following concepts: String manipulations, pointer to arrays, and pointer to functions and Structures (3 lab)
File Management: Programs related to file handling (Finding the number of characters, words and lines of given text file and File handling programs) (1 lab)</td><td></td></tr>
<tr><td colspan="6">References:</td></tr>
<tr><td>1.</td><td colspan="5">Education Solutions Limited, I. T. L. (2004). Introduction to Computer Science. India: Pearson Education.</td></tr>
<tr><td>2.</td><td colspan="5">How to Solve it by Computer, RG Dromey, PHI</td></tr>
<tr><td>3.</td><td colspan="5">The C Programming Language, Brian W. Kernighan and Dennis Ritchie, Latest Edition, Prentice Hall.</td></tr>
<tr><td>4.</td><td colspan="5">Programming in ANSI C, E. Balagurusamy, Latest Edition, McGraw Hill</td></tr>
<tr><td>5.</td><td colspan="5">Let us C, Yashavant Kanetkar, Latest Edition, BPB Publication</td></tr>
</table>

Second Semester					
S. No	**Code**	**Subject**	**L-T-P**	**Credit**	**Type**
		Institute Core Subjects		***18***	***IC***
	AIT1xx	Data Structures	3-0-0	3	PC
	AIT1xx	Mathematics for AI	3-0-0	3	PC
	AIP1xx	Data Structures Lab	0-0-2	1	PC
				25	

Data Structures					
Prerequisite: :NiL		L	T	P	C
Total hours: 42		3	0	0	3
Course Content					Hrs
Unit 1	Fundamentals of Data Structures, Memory Allocation, Abstract Data Types, Asymptotic Notations, Arrays, Lists Stack Implementation, Stack applications. Queue Implementation, Sequential, Circular, and Dequeue representation, Dynamic Queue implementation, Queue applications.				8
Unit 2	Searching and Sorting: Linear and Binary search, Bubble Sort, Selection Sort, Insertion Sort, Merge sort, Quick sort, Counting sort, Bucket sort, Radix sort, Heap sort, comparisons of sorting algorithms.				8
Unit 3	Hashing and Hash Tables: Hash functions, Open and closed hashing, Dynamic and extendible hashing, Hash collision, chaining, Hash Tables and Probing Techniques				8
Unit 4	Trees: Binary Tree and its representations, Tree traversal, Binary Search Tree, Threaded binary trees, Representing list as binary trees, Dynamic implementation of Binary tree and AVL tree, Tree applications, Interval tree, M-way search Tree, B-Tree and its variants , B+ Tree , Heaps and its applications				10
Unit 5	Graphs: Fundamentals of Graph, Adjacency Matrix and List; Graph Traversal using DFS and BFS. Dijkstra and Prims algorithms.				8

References

1.	T.Cormen, C.Lieserson, R.Rivest, and C.Stein, "Introductions to Algorithms", Prentice-Hall/India, 3rd edition, 2009
2.	Aaron M. Tenenbaum, Y. Langsam, Moshe J. Augenstein, Data Structures Using C
3.	Introduction to Algorithms ,Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein,PHI,2nd Edition.
4.	Aho A.V., J.E. Hop croft, J.D. Ullman, Data Structures and algorithms, Addison Wesley
5.	Introduction to design & Analysis of Algorithms,Anany Levitin,2ndEdition,Pearson.

Mathematics for AI				
Prerequisite: Some basic set theory (what is a set and elementary set operations), combinatorics (knowing different ways of counting, inclusion-exclusion principle) and calculus (knowing derivatives and integrals)	L	T	P	C
Total hours: 30	3	0	0	3

Course Content		Hrs
Unit 1	Linear Algebra Scalars, Vectors, Matrices and Tensors, Multiplying Matrices and Vectors , Identity and Inverse Matrices, Linear Dependence and Span, Norms, Special Kinds of Matrices and Vectors, Eigen decomposition, Singular Value Decomposition, The Moore-Penrose Pseudoinverse, The Trace Operator, The Determinant, Principal Component Analysis.	8
Unit 2	Probability and Information Theory, Random Variables, Probability Distributions, Marginal Probability, Conditional Probability, The Chain Rule of Conditional Probabilities, Independence and Conditional Independence, Expectation, Variance and Covariance, Common Probability Distributions ix. Useful Properties of Common Functions, Technical Details of Continuous Variables, Information Theory, Structured Probabilistic Models	8
Unit 3	Statistical inference: statistical decision theory, statistical assumptions, estimation theory. Methods of estimation: method of moments, method of minimum variance.	8
Unit 4	Statistical hypothesis testing, null and alternate hypotheses. Simple and composite hypotheses, Type-I and Type-II errors, Z-tests for difference of means, chi-square test, tests for correlation and regression.	6

References

1.	Linear Algebra, Gilbert Strang, MIT Cambridge Press
2.	Foundations of Learning, Julie Fisher, Open University Press
3.	Foundations of Learning, Laurie L. Hazard, Jean-Paul Nadeau, Pearson
4.	Probability and Statistics for Machine Learning, Anirban Das Gupta, Springer

Data Structures Lab					
Prerequisite: Basic Programming Skills		L	T	P	C
Total hours: 30		0	0	2	1
Course Content					Hrs
	The following topics are broad areas. The instructor offering the course in consultation with the theory offered can adopt further variations in tune with concerned theory courses.				
	Programming assignments for the conceptual understanding of control constructs, scoping rules, sparse metrics, single linked list, and multi-list. Searching: Linear Search, Binary Search, Median Search, Hash Table. Sorting: Merge, Quick, Radix, Bucket, and Count; Time and Space complexity analysis of searching and sorting algorithms.				
	Non-Linear Data Structure : Binary Tree, K-ary Tree, Binary Search Tree, Threaded Tree, AVL Tree, B Tree, B+ Tree, Priority Queue using Binary Heap. Graph: Adjacency Matrix and List; Graph Traversal using DFS and BFS				

References

1.	T.Cormen, C.Lieserson, R.Rivest, and C.Stein, “Introductions to Algorithms”, Prentice-Hall/India, 3rd edition, 2009
2.	Aaron M. Tenenbaum, Y. Langsam, Moshe J. Augenstein, Data Structures Using C
3.	Introduction to Algorithms ,Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein,PHI,2nd Edition.
4.	Aho A.V., J.E. Hop croft, J.D. Ullman, Data Structures and algorithms, Addison Wesley
5.	Introduction to design & Analysis of Algorithms, Anany Levitin, 2nd Edition, Pearson.

Scheme and Syllabus of 3rd Semester

Third Semester					
S. No	**Code**	**Subject**	**L-T-P**	**Credits**	**Type**
	22AIT2xx	Digital Systems and Computer Architecture	3-1-0	4	PC
	22AIT2xx	Design and Analysis of Algorithms	3-0-0	3	PC
	22AIT2xx	Operating Systems	3-0-0	3	PC
	22AIT2xx	Foundations of Data Science	3-0-0	3	PC
	22AIT2xx	Theory of Computation	3-1-0	4	PC
	22HST2xx	Social Sciences and Professional Ethics	3-1-0	4	BS
	22AIP2xx	Data Science Lab	0-0-2	1	PC
	22AIP2xx	Design and Analysis of Algorithms Lab	0-0-4	2	PC
	22AIP2xx	Operating Systems Lab	0-0-2	1	PC
			29	**25**	

Digital Systems and Computer Architecture					
Prerequisite: Nil		L	T	P	C
Total hours:		3	1	0	4
Course Content					Hrs
Unit 1	Boolean Algebra and Logic Gates: Basic definition, Axiomatic Definition, Basic theorem and Properties of Boolean algebra, Minterms and Maxterms, Logic Operations, Digital logic gates IC, digital logic families.				06
Unit 2	Simplification of Boolean functions: Different types map method, product of sum simplification, NAND or NOR implementation, Don't care condition, Tabulation method, Adder, subtractor, Code Conversion, Universal Gates.				06
Unit 3	Sequential Logic: Flip-flops, Triggering of Flip-flops, Analysis of clocked sequential circuits, State reduction and Assignment, Flip-flop excitation, Design of counters, Design with state equations.				06
Unit 4	Basic Computer Organization and Design: Instruction codes, Computer registers, Computer instructions, Addressing modes, Timing and Control, Instruction cycle, Memory-Reference Instructions, Input-output and interrupt. Design of Basic computer, Design of Accumulator Unit.				08
Unit 5	Programming the Basic Computer: Introduction, Machine Language, Assembly Language, the Assembler. Programming Arithmetic and logic operations. Subroutines. I-O Programming.				07
Unit 6	Central Processing Unit: Introduction, General Register Organization, Stack Organization, Instruction format. Addressing Modes, Data transfer and manipulation. Program Control, Reduced Instruction Set Computer (RISC). Pipelining, Parallel Processing.				06

References

1.	Computer System Architecture: By M. Morris Mano.
2.	Digital logic and computer design: M. Morris Mano, PHI
3.	Structured Computer Organization: By Tanenbaum.
4.	Computer Organization: By Stallings.
5.	Computer Architecture and Organization: By Hayes.
6.	Advanced Computer Architecture by Kai Hwang
7.	Microprocessor Architecture, Programming, and Applications with the 8085 Ramesh S. Gaonkar Pub: Penram International.

Design and Analysis of Algorithms					
Prerequisite: Data Structures	L	T	P	C	
Total Hours: 42	***3***	***0***	***0***	***3***	

	Course Content	Hrs
Unit 1	***Algorithm Analysis:*** Asymptotic notation, model of computation, time and space complexities, average and worst-case analysis, Master's Theorem, solving recurrence equations- interation method, substitution, recursion tree, master method. Amortised Analysis. Linear Search, Insertion Sort, Euclid's Algorithm for finding GCD (Lame's Theorem): Correctness, Best-Case, Average-Case and the Worst-Case Running Time Analysis. Permutation Model for Average-Case Analysis of an Algorithm for Finding Maximum Element in an Array	8
Unit 2	***Divide and Conquer*:** General recurrence and methods for obtaining bounds on given recurrence. Binary Search, Merge Sort, and Maximum Subarray Sum Problem. Quick-sort: Correctness, Running Time Analysis, Order statistics - finding median and Worst-case Linear Time Algorithm for Selection Problem. Max-Min problem, Strassen's Algorithm for Matrix Multiplication, Karatsuba`s Algorithm for Large Integer Multiplication	8
Unit 3	***Dynamic Programming Approach:*** Introduction to dynamic programming - principal of optimality, Optimal substructure. Matrix Chain Multiplication Problem, Optimal Binary Search Tree Problem, Longest Common Subsequence Problem, 0/1 Knapsack Problem. ***Greedy Approach:*** Elements of Greedy Strategy - Greedy choice property, optimal substructure. Example Problems - Activity Selection Problem, Fractional Knapsack Problem, Huffman codes, Travelling Salesman Problem.	9
Unit 4	***Graph Algorithms:*** Graph Traversal Algorithms (BFS, DFS), Shortest path algorithms (Bellman-ford, Dijkstra's, Transitive-Closure, Floyd-Warshall), minimum spanning tree algorithms Kruskal, Prim), Network-flow (ford-fulkerson) , applications of DFS:- bi-connectivity, topological sort, strongly-connected components, Articulation point.	9
Unit 5	***Backtracking:*** Introduction to Backtracking, Enumerating Independent Sets of a Graph, Graph Coloring Problem and N-Queen's Problem. ***Complexity Classes:*** P, NP, NP-Hard and NP-Complete. NP-Complete Examples with Reductions: Satisfiability, Clique, Independent Set, Vertex Cover , Graph Coloring, Dominating Set,	8

References

1.	Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, Introduction to Algorithms, Third Edition, PHI, 2009.
2.	Ellis Horowitz, Sartaj Sahni and Sanguthevar Rajasekaran, Fundamentals of Computer Algorithms, Second Edition, Universities Press, 2011.
3.	Michael T. Goodrich and Roberto Tamassia, Algorithm Design: Foundations, Analysis and Internet Examples, Second Edition, Wiley-India, 2006.
4.	Michael R. Garey and David S. Johnson, Computers and Intractability: A Guide the theory of NP-Incompleteness, W.H. Freeman & Co., 1979.
5.	Herbert S. Wilf, Algorithms and Complexity, AK Peters Ltd., 2003.
6.	Jon Kleinberg and Eva Tardos. 2005. Algorithm Design. Addison-Wesley Longman Publishing Co., Inc., USA.

Operating Systems					
Total Hours		L	T	P	C
42		3	0	0	3
Prerequisite: Computer Organization and Architecture, Data structures and algorithms, Problem solving using C					
Course Content					Hrs
Unit 1	***Introduction:*** What is an operating system, Types of operating systems and differences among them; Basic Computer Architecture, OS as a virtual machine; User and Operating-System Interface, System Calls, System Services, Linkers and Loaders, Booting, OS as a resource manager, Interrupts and traps, System calls, Limited direct execution, user versus kernel mode. ***CPU Scheduling:*** Process, Process v/s program, context switch, Process state diagram, CPU scheduling – FCFS, SJF, SRTF, Priority, Pre-emptive priority, Round Robin, MLFQ, Lottery, CFS, Multi-Processor Scheduling, Real-Time CPU Scheduling, Thread v/s process, Process and Thread APIs				10
Unit 2	***Synchronization:*** Inter-process communication and Processes: IPC in Shared-Memory Systems and Message-Passing Systems, Race condition, mutual exclusion, The Critical-Section Problem (CSP), Algorithmic solutions to CSP – Dekker's, Peterson's, Lamport Bakery Solution; Hardware Support for Synchronization – Test and Set, Compare and Swap; OS support for synchronization - Mutex Locks, Semaphores, Monitors; Condition Variables; Classic Problems of Synchronization – Producer Consumer, Sleeping Barber; Dining Philosopher's Problem, Deadlock – Prevention, avoidance, detection and recovery, Safe state, Banker's algorithm. Livelock.				10
Unit 3	***Memory Management:*** working set model, hardware support; Contiguous allocation-partitioned memory allocation – fixed and variable partitioning, memory management with bit maps – swapping – relocation- protection and sharing. Non contiguous allocation – Paging – principles , page allocation, segmentation. Virtual memory concepts, address translation, management of virtual memory, page replacement policies, protection and sharing, Thrashing; Caching principles and quantitative estimation of cache behavior				8
Unit 4	***I/O Management:*** Overview of Mass-Storage Structure, HDD Scheduling, NVM Scheduling, Error Detection and Correction, Storage Device Management, Swap-Space Management, SSD (Solid State Disks); I/O Systems -Overview; I/O Hardware; Kernel I/O Subsystem, Transforming I/O Requests to Hardware Operations ***File management:*** File Concept, Access Methods, Directory Structure, Protection, File-System Interface, Shared files. File-System Implementation: Structure and Operations; Directory Implementation; Allocation Methods; Free-Space Management; Case study: EXT, NTFS, HFS				8
Unit 5	***Security and Protection:*** Program Threats – stack overflow, return to libc, RoP, heap spraying, integer overflow, format string attacks; System and Network Threats; User Authentication; Principles of Protection - Protection Rings, Domains; Access Matrix, Implementation of the Access Matrix – Access Control Lists, capabilities; Revocation of Access Rights, Role-Based Access Control, Mandatory Access Control, Capability-Based Systems				6
References					

1.	Remzi H. Arpaci-Dusseau and Andrea C. Arpaci-Dusseau, *Operating Systems:Three Easy Pieces* [online http://pages.cs.wisc.edu/~remzi/OSTEP/]
2.	Abraham Silberschatz, Peter B. Galvin, Greg Gagne, *Operating System Concepts*. 9th edition. Wiley.
3.	Andrew Tanenbaum & Albert Woodhull, *Operating Systems: Design and Implementation*. Prentice-Hall.
4.	Maurice J Bach, *Design of Unix Operating System*. AT&T Bell Labs.
5.	Andrew Tanenbaum, *Modern Operating Systems*, Prentice Hall.
6.	William Stallings, *Operating Systems: Internals and Design Principles*, 9th Edition, Pearson.
7.	Crowley: *Operating System A Design Approach*, TMH.

Foundation of Data Science					
Prerequisite: Mathematics for AI	L	T	P	C	
Total Hours: 40	***3***	***1***	***0***	***4***	

Course Content		Hrs
Unit 1	Review of ProbabilityTheory: Probability basics, Descriptive statistics, Sampling and sampling distributions Point estimation and interval estimation Confidence intervals, Hypothesis testing, Overview of Data Science.	8
Unit 2	Hypothesis testing (null and alternative hypotheses), Confidence intervals, p-values and significance levels, Type I and Type II errors, Sampling and Sampling Distributions: Simple random sampling, Stratified sampling, Sampling distributions of sample statistics (e.g., sample mean, sample proportion)	8
Unit 3	Parametric and Non-parametric Tests: t-tests (independent samples, paired samples), Analysis of variance (ANOVA), Chi-squared tests, Wilcoxon signed-rank test, Mann-Whitney U test, Kruskal-Wallis test, Kolmogorov-Smirnov test	8
Unit 4	Model evaluation and interpretation, Time Series Analysis: Time series data and components, Autocorrelation and partial autocorrelation, Forecasting methods (e.g., moving averages, exponential smoothing, ARIMA), Bayesian Statistics: Bayes' theorem, Bayesian inference, Markov Chain Monte Carlo (MCMC) methods, Machine Learning and Statistics: Evaluation metrics (e.g., accuracy, precision, recall, F1-score).	8
Unit 5	Data Visualization: Creating effective plots and charts, Data exploration and storytelling through visualization, Ethics and Data Science: Ethical considerations in data collection and analysis, Bias and fairness in machine learning	8

References

1.	Cathy O’Neil and Rachel Schutt, “ Doing Data Science, Straight Talk From The Frontline”, O’Reilly, 2014.
2.	Joel Grus, “Data Science from Scratch: First Principles with Python”, O’Reilly Media, 2015.
3	Wes McKinney, “Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython”, O’Reilly Media, 2012.

Theory of Computation				
Prerequisite: NIL	L	T	P	C
Total hours: 40	3	0	0	3
Course Content				Hrs
Unit 1	**BASIC FOUNDATION**: Review Of SET Theory, Automata Theory, Alphabet, Power Of Alphabet, Kleen Closure, Positive Closure, String, Empty String, Concatenation, Language. **FINITE AUTOMATA (FA):** Introduction, Deterministic Finite Automata (DFA) -Formal Definition, Simpler Notations (State Transition Diagram, Transition Table), Language of A DFA. Nondeterministic Finite Automata (NFA)- Definition of NFA, Language of an NFA, Equivalence Of Deterministic and Nondeterministic Finite Automata, Applications of Finite Automata, Finite Automata with Epsilon Transitions, Eliminating Epsilon Transitions, Minimization Of Deterministic Finite Automata, Finite Automata with Output (Moore and Mealy Machines) and Inter Conversion.			8
Unit 2	**REGULAR EXPRESSIONS (RE):** Introduction, Identities of Regular Expressions, Finite Automata and Regular Expressions- Converting from DFA's to Regular Expressions, Converting Regular Expressions to Automata, Minimization of Finite Automata, Applications of Regular Expressions. **REGULAR GRAMMARS:** Chomsky Classification of Languages, Regular Grammars and FA, FA for Regular Grammar, Regular Grammar for FA. Proving Languages to be Non-Regular -Pumping Lemma, Applications, Closure Properties of Regular Languages.			8
Unit 3	**CONTEXT FREE GRAMMER (CFG):** Derivation Trees, Sentential Forms, Rightmost and Leftmost Derivations of Strings. Ambiguity in CFG's, Minimization of CFG's, Normal Forms (CNF, GNF), Pumping Lemma for CFL's			8
Unit 4	**PUSHDOWN AUTOMATA THEORY:** Push Down Automata, Deterministic and Nondeterministic PDA, PDA And Languages, Construction of PDA, Acceptance of CFL, Acceptance by Final State and Acceptance by Empty Stack and its Equivalence, Equivalence of CFG and PDA. **TURING MACHINES (TM):** Formal Definition and Behaviour, Languages of a TM, TM as Accepters, TM as a Computer of Integer Functions, TM with Storage in its State, TM as Subroutine, Minskey's Theorem, Types of TMs, Multitrack, Mutitape, Nondeterministic, TM, Encoding of TM, Computability and Acceptability.			8
Unit 5	**RECURSIVE AND RECURSIVELY ENUMERABLE LANGUAGES (REL):** Properties of Recursive and Recursively Enumerable Languages. **UNDECIBILITY And UNDECIDABLE Problems:** Post's Correspondence Problem (PCP), Universal Turing Machine, The Halting Problem, Undecidable Problems about TMs. Context Sensitive Language and Linear Bounded Automata (LBA), Chomsky Hierarchy, Decidability			8
References				
1.	John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman (2007), Introduction to Automata Theory Languages and Computation, Pearson Education, India.			
2.	Cohen, Introduction to Computer Theory, Addison Wesley.			
3.	Martin, Introduction to Languages and Theory of Computation, TMH.			
4.	Papadimitriou, Introduction to Theory of Computing, Prentice Hall.			
5	P. Linz, An Introduction to Formal Languages and Automata, Jones & Bartlett, 2016.			

Social Sciences and Professional Ethics				
Prerequisite: Nil	L	T	P	C
Total hours: 56	3	1	0	4
Course objectives: ● Augmenting the understanding of society, societal issues and problems ● To provide the students an insight into the multifaceted economic and financial environment ● Development of a positive character, empathetic human being, responsible citizen ● Inculcating a positive work culture respecting professional ethics				
Course Content				Hrs
Unit 1	**Introducing Sociology:** Meaning, scope and evolution of Sociology, Key theoretical trajectories. Society, community, Social Institutions, Social Groups, Socialisation and Culture, Norms and Values, Agency and structure			10
Unit 2	**Social Change:** Social Change, development and progress; Globalisation, Industrialisation, urbanisation and modernisation; Social mobility and social stratification			8
Unit 3	**Social Issues:** Science technology and society; Digital divide, Appropriate technology, Gender inequality; Substance abuse, Consumerism, Environmental degradation and climate crisis, Nation building			10
Unit 4	**Socio-economic environment:** Overview of Socio-economic policy environment; PESTLE analysis. Economic growth & development; primary, secondary and tertiary sectors; structural changes & emerging sectors of the Indian economy. Design and strategy of economic reforms and liberalization: India's growth post liberalization.			10
Unit 5	**Finance and banking:** Banking and Financial Sector; Reforms & Challenges; Monetary & Fiscal Policies; meaning, importance & instruments. Global economic environment and opportunities. Intellectual property rights and R & D environment.			6
Unit 6	**Ethics and values:** Professional Ethics: Need, importance and principles of Professional ethics, Ethics in relation with use of technology and technology development, diversity inclusion and equity; Social responsibility. Constitutional values: Preamble and DPSP, Rights and duties			12
References				
1.	Haralambos, Michael & Holborn, Martin. Sociology: Themes and Perspective, Harper Collins. Eighth edition. 2014.			
2.	Ritzer, George. Sociological Theories, McGraw-Hill; Fifth edition. 2011			
3.	Lillie, William. An introduction to Ethics Allied Publishers Pvt. Ltd.; 1st edition (1967)			

4.	Lama, Dalai. Ethics for the New Millennium by the. Riverhead Books; Reissue edition (2001)
5.	Uma Kapila, Indian Economy Performance and Policies, Academic Foundation, New Delhi
6.	Ahluwalia, I.J. & IMD Little, India's Economic Reform and Development, Oxford University Press.

Data Science Lab				
Pre-requisite: C Programming, Data Structures	L	T	P	C
	0	0	3	2

Course Content

1. Implementation in Python: Environment set-up, Jupyter overview, Python Numpy, Computation on NumPy Arrays
2. Basics of NumPy-Computation on NumPy-Aggregations-Computation on Arrays-Comparisons, Masks and Boolean Arrays-Fancy Indexing-Sorting Arrays-Structured Data: NumPy's Structured Array
3. Data Manipulation with Pandas, Matplotlib, Scikit tool
4. Data processing, Implement different techniques to analyze dataset. Data Indexing and Selection
5. Operations on Data, Handling Missing Data
6. Vectorising different operations on Data. High-Performance Pandas: eval() and query().
7. Implement and analysis important statistical methods on a given data used in data science using python
8. Basic functions of matplotlib-Simple Line Plot, Scatter Plot-Density and Contour Plots
9. Histograms, Binnings and Density-Customizing Plot Legends, Colour Bars-Three-Dimensional Plotting in Matplotlib
10. Data visualization: Tableau. Creating charts, Mapping data in Tableau

References

1	Jake VanderPlas ,Python Data Science Handbook - Essential Tools for Working with Data, O'Reily Media, Inc, 2016
2	Joel Grus ,Data Science from Scratch First Principles with Python, O'Reilly Media,2016
3	T.R Padmanabhan, Programming with Python,Springer Publications,2016.

Design and Analysis of Algorithms Lab					
Pre-requisite: C Programming, Data Structures		L	T	P	C
		0	0	2	1
Course Content					
	1. Implementation of various sorting and searching algorithms (Revision) 2. Implement quick sort with three different positions of pivot element- first, last, random 3. Implement Tree traversal, and graph traversal (recursive algorithms) 4. Implement deterministic and randomized selection problem 5. Implement maximum subarray sum problem 6. Implement Karatsuba`s Algorithm for Large Integer Multiplication 7. Implement matrix chain multiplication, longest common sub-sequences, 0/1 knapsack 8. A program to obtain the topological ordering of vertices in a given digraph. 9. Implement travelling salesman problem. 10. Print all the nodes reachable from a given starting node in a digraph using BFS method. 11. Check whether a given graph is connected or not using DFS method. 12. Find minimum cost spanning tree of a given undirected path using a Prim's algorithm. 13. From a given vertex in a weighted connected graph, find shortest paths to other vertices using Dijkstra's algorithm.				
References					
1	Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, Introduction to Algorithms, Third Edition, PHI, 2009.				
2	Ellis Horowitz, Sartaj Sahni and Sanguthevar Rajasekaran, Fundamentals of Computer Algorithms, Second Edition, Universities Press, 2011.				
3	Michael T. Goodrich and Roberto Tamassia, Algorithm Design: Foundations, Analysis and Internet Examples, Second Edition, Wiley-India, 2006.				

Operating Systems Lab				
	L	T	P	C
Number of Weeks: 14	0	0	2	2
Pre-requisite: C Programming, Linux basics, Python				

Course Content

1) Write a C/Python program to simulate CPU scheduling. Following CPU scheduling mechanisms need to be implemented:
   a) SJFS, FCFS
   b) Priority (pre-emptive & non-pre-emptive)
   c) Round Robin
   d) MLFQ
   e) Lottery
2) Given a list of process IDs, write a program to develop a tree depicting ancestor/parent/child relationship. This shall be a dynamic scenario, and the tree should be updated every second (as new child processes may be created and some may be killed or terminated normally/abnormally).
3) Given two processes P1 and P2 (created as parent/child process through fork/ two threads within same process or two independent processes through two different programs) both of which increment a shared variable, implement Dekker's & Peterson's solutions.
4) Implementation of Lamport-Bakery solution for (N>= 5) processes. Each process shall increment a shared counter by one.
5) Modify solution to producer-consumer problem so that it works wherein producer produces one item but consumer consumes two items. If buffer has only one item, consumer relinquishes critical section and waits till there are two or more items. The solution should be
   a) threads based
   b) independent process based
6) Write a program to check if there is a deadlock in the resource-allocation graph. If not, how can the process be allocated resources with no deadlock ever occurring.
7) Implement Sleeping Barber and dining Philosophers problem using semaphores.
8) Write a program in C that reads a file from the file system and displays its contents on the screen. Implement error handling and permission checking.
9) Write a program in C that implements a simple memory allocation algorithm such as first-fit or best-fit, and tests its performance using a benchmark program.
10) You are given a file named "input.txt" that contains parameters related to a disk in the first six lines - number of cylinders (track), number of sectors, bytes per sector, RPM, average seek time, initial head position. These parameters are in different lines of the same file. Track 0 is the outermost one.
    The seventh line of the file should contain a sequence of requests for track (cylinder) numbers. Write a program to output
    a) Average Rotational delay
    b) Total Seek Time to service all the requests for
       - SSTF (Shortest Seek time first)
       - LOOK
11) Create a virtual machine using Virtual Box or VMware, install an operating system on it, and configure it to run a web server. Test the web server using a web browser and network analysis tools
12) Implement buffer overflow attack using stack smashing.
13) Write a shell script that performs the following tasks:
    a) File manipulation: Create, delete, copy, and move files and directories.
    b) Text processing: Search for specific patterns in files and perform text transformations.

c) System monitoring: Retrieve system information like CPU usage, memory utilization, and disk space.
d) Automation: Automate a repetitive task on your Linux system using a shell script.

14) Implement a program in Linux that demonstrates the following process management concepts:
    a) Process creation: Create child processes using the fork() system call.
    b) Process termination: Terminate processes using the exit() system call.
    c) Process synchronization: Synchronize processes using semaphores, mutexes, or other synchronization mechanisms.
    d) Signal handling: Handle signals like SIGINT or SIGTERM in your program.
15) Develop a program that interacts with the Linux file system. Your program should enable users to:
    a) Create files and directories.
    b) Navigate through directories and display their contents.
    c) Copy or move files and directories.
    d) Change file permissions and ownership.
16) Write a simple Linux device driver that interacts with a custom hardware device or simulates a virtual device. Your device driver should:
    a) Implement read and write operations to interact with the device.
    b) Handle interrupts or other device-specific functionalities.
    c) Test the device driver by accessing the device and performing read/write operations.

References

1.	Remzi H. Arpaci-Dusseau and Andrea C. Arpaci-Dusseau, *Operating Systems:Three Easy Pieces* [online http://pages.cs.wisc.edu/~remzi/OSTEP/]
2.	Abraham Silberschatz, Peter B. Galvin, Greg Gagne, *Operating System Concepts*. 9th edition. Wiley.
3.	Andrew Tanenbaum & Albert Woodhull, *Operating Systems: Design and Implementation*. Prentice-Hall.
4.	Maurice J Bach, *Design of Unix Operating System*. AT&T Bell Labs.
5.	Andrew Tanenbaum, *Modern Operating Systems*, Prentice Hall.
6.	William Stallings, *Operating Systems: Internals and Design Principles*, 9th Edition, Pearson.
7.	Crowley: *Operating System A Design Approach*, TMH.

Scheme and Syllabus of 4th Semester

Fourth Semester					
S. No	**Code**	**Subject**	**L-T-P**	**Credits**	**Type**
	22AIT2xx	Artificial Neural Networks	3-0-0	3	PC
	22AIT2xx	Artificial Intelligence	3-0-0	3	PC
	22AIT2xx	Compiler Design	3-0-0	3	PC
	22AIT3xx	Computer Networks	3-0-0	3	PC
	22AIT2xx	Database Management Systems	3-0-0	3	PC
	22MMTxx	Basics of Management	3-0-0	3	PLEAS
	22AIP3xx	Computer Networks Lab	0-0-4	2	PC
	22AIP2xx	Artificial Intelligence Lab	0-0-4	2	PC
	22AIP2xx	Database Management Systems Lab	0-0-2	1	PC
	22AIP2xx	**Open-ended Minor Project**	**0-0-4**	**2**	**PC**
			32	**25**	

Artificial Neural Networks				
Prerequisite: Basic understanding of probability and statistics, linear algebra and calculus. A basic knowledge of programming (preferably Python) is essential	L	T	P	C
Total hours: 42	3	0	0	3

Course Content		Hrs
Unit 1	Introduction to Artificial Neural Networks : Introduction, Artificial Neural Networks, Historical Development of Neural Networks, Biological Neural Networks, Comparison Between them and the Computer, Comparison Between Artificial and Biological Neural Network Basic Building Blocks of Artificial Neural Networks, Artificial Neural Network (ANN) terminologies.	10
Unit 2	Fundamental Models of Artificial Neural Networks : Introduction, McCulloch - Pitts Neuron Model, Learning Rules, Hebbian Learning Rule Perceptron Learning Rule, Delta Learning Rule (Widrow-Hoff Rule or Least Mean Square(LMS)Rule,Competitive Learning Rule, Out Star Learning, Boltzmann Based Learning, Hebb Net. Perceptron Networks : Introduction, Single Layer Perceptron, Brief Introduction to Multilayer Perceptron Networks.	10
Unit 3	Associative Memory Networks: Introduction, Algorithms for Pattern Association, Hetero Associative Memory Neural Networks, Auto Associative Memory Network, Bi-directional Associative Memory.	12
Unit 4	Feedback Networks: Introduction, Discrete Hopfiled Net, Continuous Hopfiled Net, Relation between BAM and Hopfiled Nets. Feed Forward Networks: Introduction, Back Propagation Network (BPN), Radial Basis Function Network (RBFN). Self Organizing Feature Map : Introduction, Methods Used for Determining the Winner, Kohonen Self Organizing Feature Maps, Learning Vector Quantization (LVQ),Max Net, Maxican Hat, Hamming Net	12

References

1.	S. Haykin, “Neural Networks and Learning Machine”s , 3rd Edition , Prentice-Hall , 2008 , ISBN No. 0131471392
2.	Jacek M. Zurada, “Introduction to Artificial Neural Systems , Jaico Publishing House; First edition.
3.	B Yegnanarayana, “Artificial neural networks”, 1st ed., Prentice Hall of India P Ltd, 2005.

Artificial Intelligence					
Prerequisite: Some basic set theory (what is a set and elementary set operations), logic, probability, and continuous mathematics	L	T	P	C	
Total Hours: 42	***3***	***0***	***0***	***3***	

Course Content		Hrs
Unit 1	Introduction: What is AI, Foundation of AI and its history, Agents and Environment	8
Unit 2	Problem Solving: Solving problem by searching, Beyond classical search, Adversarial search, Constraint satisfaction problems	8
Unit 3	Knowledge, reasoning and planning: Logical agents, First order logic, Inference in First order logic, Knowledge representation.	9
Unit 4	Uncertain knowledge and reasoning: Quantifying uncertainty, Probabilistic reasoning, Probabilistic reasoning overtime, Inference in temporal models, Hidden Markov models, The basis of utility theory, Utility functions, Multi-attribute utility functions	9
Unit 5	Learning: Learning from examples, Evaluating and choosing the best hypothesis, The theory of learning, Knowledge in learning	8

References

1.	Artificial Intelligence a Modern Approach, III Edition, Stuart Russell and Peter Norvig
2.	Elaine Rich, Kevin Knight, & Shivashankar B Nair, Artificial Intelligence, McGraw Hill, 3rd ed.,2009
2.	Probability and Statistics for Machine Learning, Anirban Das Gupta, Springer
3.	The Elements of Statistical Learning, Trevor Hastie, Robert Tibshirani, second ed, Springer

Compiler Design					
Prerequisite: Theory of Computation		L	T	P	C
Total hours: 42		3	0	0	3
Course Content					Hrs
Unit 1	Language Translators: Compilers and Interpreters, Hybrid Compiler, Structure of a Compiler, Self Compiler and Cross Compiler. Lexical Analysis: Design and implementation of Lexical Analyzers, Finite automata and Regular expressions, Lex tool – the Lexical Analyzer Generator.				8
Unit 2	Syntax Analysis: Context Free Grammars, Derivation and Parse trees, Ambiguity of grammars. Bottom-up and Top-down Parsing - Shift Reduce Parser, Operator Precedence Parser, First and Follow functions, Left recursion, LL Parsers, Canonical collection of items, LR parsers, Conflict Resolution in LR parsers.				14
Unit 3	Syntax-Directed Translation: Syntax-directed definitions and translation schemes, Attributes and Translation Rules, Implementation of S-attributed and L-attributed definitions. Intermediate Code Generation: Intermediate codes, Three address codes, Translation of Expressions and Type Checking.				8
Unit 4	Code Optimization and Code Generation : Basic blocks, Flow graphs, DAG, Global data flow analysis, ud-chaining, Available expressions, Loop optimization, Compilation of Expression and Control structures. Error Detection and Recovery.				12

References

1.	Aho, Lam, Sethi and Ullman: Compilers – Principles, Techniques and Tools, Pearson Education
2.	Tremblay and Sorenson: The Theory and Practice of Compiler Writing, BS Publications.
3.	Allen Holub : Compiler Design in C, Prentice Hall India.

Computer Networks					
Prerequisite: Fundamental knowledge on signals and systems, basics of linear algebra and calculus, and programming skills		L	T	P	C
Total hours: 42		3	0	0	3
Course Content					Hrs
1.	**Internetworking and Routing-I:** Computer Network Architecture, Circuit switching, Packet And Massage Switching, Network Structure. OSI 7-layer architecture and TCP/IP architecture. Physical Layer, network programming, Data Link Layer, Framing, Error detection. (10 Classes)				
2.	**Internetworking and Routing-II:** Retransmission algorithms. Stability of queuing systems. Multiple access and Aloha. CSMA/CD and Ethernet. High Speed LANs and Token Ring. High speed switches scheduling, IPv4 and IPv6, Broadcast routing and spanning trees. Shortest path routing. Distributed routing algorithms, optimal routing, and traffic engineering. (10 Classes)				
3.	**Resource Sharing:** Queuing models and introduction to Little's theorem, M/M/1 and M/M/m queues. Network of queues. Introduction to M/G/1 queues, reservations and priority. (8 Classes)				
4.	**End-to-End protocols and Applications:** Flow control – window/credit schemes, rate control schemes, Transport layer and TCP/IP. Introduction to ATM networks and Network Management And Interoperability. Performance Issues Of LAN And WAN. Application layer: Domain Name System (DNS), HTTP, FTP, E-mail, www and etc (9 Classes)				
5.	**Future/Advanced Internet:** Internet of Things (IoT) and applications, Software Defined Networks (SDN) : Control plane, data-plane, and issues, Information centric networks (ICN), Content distribution networks (CDN) and Future Internet.(5 Classes)				

References	
1.	Data Networks: Bertsekas and Gallagher, PHI
2.	Computer Networks: L. Peterson and Davie, Elsevier
3.	Computer Networking A top down Approach: J.F.Kurose, Pearson.
4.	Computer Networks : Andrew S. Tanenbaum, Pearson

Database Management Systems				
Prerequisite: Data Structures	L	T	P	C
Total hours: 40	3	0	0	3

Course Content		Hrs
Unit 1	Introduction to Database System Database approach and Information systems , Database System Architecture, current advances in database technology, Database Systems Development Life Cycle- Prototyping methodology three-schema architecture, three- tiered architecture Hierarchical model, Network model, Relational model, Object oriented model, Multidimensional model	6
Unit 2	Database Models: ER-model notation, entity & entity type, relationship & relationship type, Degree, Cardinality & modality, Supertype/Subtype relationship Relational model concepts, Converting ER to Relational model	6
Unit 3	Introduction to SQL-DDL,DML and DCL, Advanced topics of SQL, PL/SQL language: Functions, Procedures & triggers, Views, Cursors etc. Formal query languages Relational Algebra and Relational Calculus Overview, Query processing and optimization	10
Unit 4	Relational schema, Functional dependencies, Inference axioms, Keys, closures, redundant FD's , Decompositions, Join Dependencies Normalization, normal forms:1NF, 2NF, 3NF, BCNF, 4NF, 5NF, Best Database Design criterion Transactions, concurrency control, Crash Recovery, Physical DB design, file organizations, Indexing Structures, File indexing, hashing	14
Unit 5	Client/Server database architecture Application Development, Database Security, Overview of Distributed database, Data Warehousing and Data mining, Data Analytics	4

References

1.	**Database System Concepts** ,Silberschatz A, Korth H F, and Sudarshan S, , McGraw Hill,,6th Ed.
2.	**Modern Database Management systems**, Hoffer J A, Prescott M B, and Topi H. Pearson Education Inc.,13th Edition
3.	**Fundamentals of Database Systems**, Elmasri R, Navathe S B, Pearson Education, 7th Edition..
4.	**Database Management System**, Raghurama krishnan & Johannes Gehrke, McGraw-Hill 3rd edition
5	**Commercial Application development using ORACLE Developer 2000 Forms 5.0**, Ivan Bayross, BPB Publications.

Basics of Management				
Department: Department of Management Studies	L	T	P	C
Prerequisite: None	3	0	0	3

Course Learning Objectives

By the end of this course student will be able to:

1. Demonstrate the roles, skills and functions of managers.
2. Develop the understanding and cognizance of the importance of management principles.
3. Make effective application of acquired knowledge to diagnose and solve organizational problems and develop optimal managerial decisions.
4. Understand seven Ps of marketing and digital marketing strategies.
5. Get to know about key people management processes.
6. Understand the decisions and processes in operations management.
7. Gain knowledge of financial systems, institutions, regulators, and instruments.
8. Diagnose and communicate the complexities associated with management of various issues in the organizations and integrate the learning in handling these complexities

Course Content

1	General Management Processes and Principles: - Concept, Functions and Principles of Management, Roles and skills of Managers.	8
2	Functions of Management: Planning, Decision Making; Organizing: Organizational Design & Organizational Structures; Leading, Motivation, Communication and Controlling;	10
3	Introduction to Human Resource and Marketing Management: Trends and Practices in People Management; Marketing Management Process and decisions, Marketing Mix;	12
4	Introduction to Finance and Operations Management: Overview of Financial Systems, Financial Institutions, Markets, and Instruments; Decisions & processes in Operations Management.	10

References

1.	Robbins, Stephen P. and Coulter, Mary (2019) 'Management', 14th edition, Prentice Hall of India
2.	Dessler, G. & Varkkey, B. (2018). Human Resource Management, 15e, Pearson.
3.	Laasch, O. (2021). Principles of Management-Practicing Ethics, Responsibility, Sustainability, 2nd Edition, Sage Publications.
4.	Hill, Charles W L and McShane, Steven L. (2017) Principles of Management, Special Indian Edition, McGraw Hill Education
5.	Khan, M. Y. and Jain P. K. (Latest edition). Financial Management, Text, Problems & Cases. Tata McGraw Hill Company, New Delhi.
6.	Philip Kotler. (Latest edition). Marketing Management: Analysis, Planning, Implementation & Control. Prentice Hall of India.
7.	Koontz, Harold and Weihrich, Heinz & Ramachandra Aryasri A. (2016). Principles of Management, Latest edition, McGraw Hill Education

Computer Networks Lab					
Prerequisite: The programming lab in C++, which means you need to be very comfortable with C++ and using standard debugging tools.		L	T	P	C
Total hours: 36		0	0	4	2
Course Content					Hrs
	The laboratory experiments conducted on various tools Lab 1-3: Introduction networking (wireshark,, TCP dump, CISCO packet tracer) Lab 3-4: Introduction to socket programming Lab 5-9: Experiments on NS2 and NS3 Lab 10-12 : Experiments Mininet				48
References					
1.	Data Networks: Bertsekas and Gallagher, PHI				
2.	Computer Networks: L. Peterson and Davie, Elsevier				
3.	Computer Networking A top down Approach: J.F.Kurose, Pearson				
4.	Computer Networks : Andrew S. Tanenbaum, Pearson				

<table>
<tr><td colspan="6">Artificial Intelligence Lab</td></tr>
<tr><td colspan="2">Prerequisite: NiL</td><td>L</td><td>T</td><td>P</td><td>C</td></tr>
<tr><td colspan="2">Total hours: 28</td><td>0</td><td>0</td><td>4</td><td>2</td></tr>
<tr><td colspan="5">Course Content</td><td>Hrs</td></tr>
<tr><td>1</td><td colspan="4">Uninformed Search Algorithms in Artificial Intelligence:
I. Depth First Search (DFS)
Problem: Implement the distinct island problem using Depth First Search (DFS)
II. Breadth First Search (BFS)
Problem: Implement water jug problem using Breadth First Search (BFS)</td><td>4</td></tr>
<tr><td>2</td><td colspan="4">Informed Search Techniques:
I. Implement 8-Puzzle Problem using Hill Climbing
II. Implement 8-Puzzle Problem using Best First Search.
III. Implement Tic Tac Toe using Minimax algorithm.
IV. Implement 8 Queens Problem with Best First Search
V. Implement 8-puzzle problem using A* Algorithm
VI. Implement N-Queens problems
VII. Define a model (PEAS) for a Wumpus world and solving it.</td><td>14</td></tr>
<tr><td>3</td><td colspan="4">Constraint Satisfaction Problem: Implement Crypt Arithmetic Problem.
Model finding and applying inference algorithms like forward and backward chaining.</td><td>5</td></tr>
<tr><td>4</td><td colspan="4">Computing probability and joint distributions given a Bayesian network and associated CPTs.</td><td>5</td></tr>
<tr><td colspan="6">References</td></tr>
<tr><td>1.</td><td colspan="5">Artificial Intelligence: A Modern Approach by Russel and Norvig, Third Edition, Pearson, 2015.</td></tr>
<tr><td>2.</td><td colspan="5">Artificial intelligence: Concepts and Applications: Lavika Goel, Wiley Publications, 2021.
Link:https://www.amazon.in/Artificial-Intelligence-Applications-Lavika-Goel/dp/8126519932</td></tr>
<tr><td>3.</td><td colspan="5">Artificial Intelligence: Elaine Rich, Kevin Knight, Mc-Graw Hill.</td></tr>
<tr><td>4.</td><td colspan="5">Introduction to AI & Expert System: Dan W. Patterson, PHI.</td></tr>
</table>

Database Management Systems Lab					
Prerequisite: :NiL		L	T	P	C
Total hours: 35		0	0	3	2
Course Content					Hrs
I	Design exercises and various Tools of designing the ER diagram and its mapping to relational model				6
II	Programming exercises on SQL –Detailed DDL commands and queries to create databses.				6
III	Programming exercises on SQL –Detailed DML commands				9
IV	Programming exercises on SQL –Detailed DCL commands				3
V	Programming Exercise on advanced topics of SQL, PL/SQL language : Functions, Procedures, triggers, Views, Cursors etc. Programming Exercise on advanced topics of NO-SQL, MongoDB and Cassandra				6
	There will be as semester Mini-Group Project on theme of Database Information system				5

References	
1.	**Database System Concepts** ,Silberschatz A, Korth H F, and Sudarshan S, , McGraw Hill,,6th Ed.
2.	**Modern Database Management systems**, Hoffer J A, Prescott M B, and Topi H.,Pearson Education Inc.,13th Edition
3.	**Fundamentals of Database Systems**, Elmasri R, Navathe S B, Pearson Education, 7th Edition..
4.	**Database Management System**, Raghuramakrishnan & Johannes Gehrke, McGraw-Hill 3rd edition
5	**Commercial Application development using ORACLE Developer 2000 Forms 5.0**, Ivan Bayross, BPB Publications.

<table>
<tr><th colspan="6">Open-ended Minor Project</th></tr>
<tr><td colspan="2">Prerequisite: Foundation of Data Science, Mathematics for AI, Programming</td><td>L</td><td>T</td><td>P</td><td>C</td></tr>
<tr><td colspan="2">Total hours: 48</td><td>0</td><td>0</td><td>4</td><td>2</td></tr>
<tr><td colspan="5">Course Content</td><td>Hrs</td></tr>
<tr><td></td><td colspan="4">The objective of this course is to impart and implement practical knowledge by the students in the area of artificial intelligence and data engineering.

In this course the students are expected to propose and implement model projects to solve real world challenges with the help of tools in the domain of artificial intelligence and data engineering.</td><td>48</td></tr>
<tr><td colspan="6">References</td></tr>
<tr><td>1.</td><td colspan="5">Latest Research Articles based on the project proposals</td></tr>
</table>

Scheme and Syllabus of 5th Semester

Fifth Semester					
S. No	**Code**	**Subject**	**L-T-P**	**Credits**	**Type**
	22AIT3xx	Digital Image Processing	3-0-0	3	PC
	22AIT2xx	Machine Learning	3-0-0	3	PC
	22AIT3xx	Big Data Analytics	3-0-0	3	PC
	22AIT2xx	Information Retrieval	3-0-0	3	PC
	22AIT3xx	Data Mining and Warehousing	3-0-0	3	PC
	22AIT3xx	Program Elective-1	3-0-0	3	PE
	22AIP3xx	Digital Image Processing Lab	0-0-2	1	PC
	22AIP2xx	Machine Learning Lab	0-0-4	2	PC
	22AIP3xx	BigData Analytics lab	0-0-2	1	PC
			26	**22**	

Honors					
	22AITxxx	Bio Medical Image Analysis		3	
	22AITxxx	Social Network Analysis		3	
				6	

Minor AIDE					
	22AIT1xx	Data Structures	3-0-0	3	
	22AIT2xx	Foundations of Data Science	3-0-0	3	

Digital Image Processing					
Prerequisite: Fundamental knowledge on signals and systems, basics of linear algebra and calculus, and programming skills		L	T	P	C
Total hours: 42		3	0	0	3
Course Content					Hrs
Unit 1	Introduction to Digital Image Processing: Digital Image Representation, Fundamental Steps in DIP, Elements of Visual Perception, Image Sensing and Acquisition, Image Model, Sampling, Quantization, Basic Relationship Between the Pixels				6
Unit 2	Image Transforms: Discrete Fourier Transform (DFT), Properties of 2D DFT, Fast Fourier Transform, Inverse FFT, Discrete Cosine Transform and KL Transform, Discrete wavelet Transform, Convolution and Correlation				8
Unit 3	Image Enhancement: Spatial Domain- Basic Gray Level Transformations, Histogram processing, Smoothing and Sharpening Spatial Filters Frequency Domain- Smoothing and Sharpening frequency domain filters, Homomorphic filtering				8
Unit 4	Image Restoration: Overview of Degradation models, Unconstrained and constrained restorations, Inverse Filtering, Wiener Filter				6
Unit 5	Image Segmentation: Detection of discontinuities, edge linking and boundary detection, thresholding, region oriented segmentation Image Compression: Need for data compression, image compression models, loss-less and lossy compression				8
Unit 6	Representation and Description: Representation schemes, boundary descriptors, regional descriptors. Morphology: Dilation, erosion, opening, closing, Hit-or-Miss Transform, some basic morphological algorithms				6

References

1.	Rafael C. Gonzalez, Richard E. Woods, Digital Image Processing, Pearson , 3rd Edition, 2008
2.	Castleman. Digital Image Processing. Prentice Hall.
3.	Anil K. Jain, Fundamentals of Digital Image Processing, Pearson , 2002

Machine Learning					
Prerequisite: Basic understanding of probability and statistics, linear algebra and calculus. A basic knowledge of programming (preferably Python) is essential.		L	T	P	C
Total hours: 42		3	0	0	3
Course Content					Hrs
Unit 1	The learning problem – learning versus design, types of learning- supervised, unsupervised, reinforcement and other views of learning. Learning Modeling: A least squares approach, linear modeling, making predictions, vector/matrix notation, linear regression, nonlinear response from a linear model				6
Unit 2	Training versus Testing: theory of generalization, interpreting the generalization bound. Generalization and over fitting: when does over fitting occur? Regularization, validation, cross validation. Bias-variance tradeoff. The Linear model: Linear classification, perceptron learning, linear regression, gradient descent, batch and stachastic gradient descent, convex functions, logistic regression, non linear transformation.				8
Unit 3	Generalization and Overfitting: when does overfitting occur? Regularization, validation Generative vs discriminative models Supervised learning – Probability review, Bayes classifier, Naive Bayesian, MAP, MLE, K- nearest neighbors, decision trees, neural networks, SVM (Linear)				16
Unit 4	Unsupervised learning – the general problem, hierarchical and partitional clustering, K-means clustering, density based clustering				8
Unit 5	Assessing classification performance – accuracy, sensitivity, specificity, the area under the ROC curve, confusion matrices, FAR, TPR, TNR, FRR, precision and recall				4

References

1.	A first course in Machine learning, Simon Rogers and mark Girolami, CRC Press
2.	Learning from Data, Yaser S Abu-Mostafa, AML books
3.	Machine learning, Marsland, CRC press
4.	An Introduction to Machine Learning, Kubat Miroslav, Springer

Big Data Analytics					
Prerequisite: Database Management System, Operating System		L	T	P	C
Total hours: 42		3	0	0	3
Course Content					Hrs
1	Overview of Database Management Systems, Introduction to Big Data, Distributed file system, Big Data and its importance, Five Vs, Drivers for Big data, Big data analytics. Apache Hadoop & Hadoop Eco-System, Hadoop Distributed File System (HDFS), YARN, MapReduce Programming Model, Spark and in memory computation, RHadoop, Data Serialization, Zookeeper				10
2	Overview of Apache Pig, Pig's role in the Hadoop ecosystem, Pig Latin syntax and data model, Loading and storing data, Grouping and joining operations, User-defined functions (UDFs), Hive data types and schemas, Hive metastore: tables, databases, and partitions, Hive Query Language (HiveQL), Writing HiveQL queries, Filtering and sorting data, Aggregating and grouping data.				8
3	NoSQL databases, Understanding the need for NoSQL databases, CAP Theorem, ACID vs BASE, Types of NoSQL databases, Overview of MongoDB and its use cases, Querying MongoDB, Inserting, updating, and deleting documents, Hbase, Hbase Internals, Cassandra data model: keyspace, tables, and columns, Data visualization techniques, Tools for data visualization (e.g., Tableau, Matplotlib)				10
4	Supervised and unsupervised learning, Machine Learning for Big Data, Mahout, Parallel K-means algorithm, Parallel and Distributed Swarm Computation, Spark Machine Learning (MLlib) introduction, Practical applications and case studies Graph Processing, Pregel, Giraph, Neo4j, Spark GraphX, Titan.				8
5	Text data Preprocessing Techniques for Massive Data, Streaming for Fast Data, Kafka, Spark Streaming, NLP techniques for Large Scale Data, Sentiment analysis and text classification, Data privacy and ethics in Big Data Security challenges and solutions, Compliance and legal considerations				6

References

1.	"Hadoop: The Definitive Guide" by Tom White, O'Reilly media, 2021.
2.	"Big Data Analytics" by V. Rajaraman and A. Konar: Elsevier Science, 2016.
3.	Joshua N. Milligan, "Learning Tableau 2020: Create effective data visualizations, build interactive visual analytics and transform your organization," Packt Publishing Limited, 2020.
4.	Nathan Marz, James Warren: Big Data: Principles and best practices of scalable realtime data systems, 2020.

<table>
<tr><th colspan="6">Information Retrieval</th></tr>
<tr><td colspan="2">Prerequisite: nil</td><td>L</td><td>T</td><td>P</td><td>C</td></tr>
<tr><td colspan="2">Total hours: 40</td><td>3</td><td>0</td><td>0</td><td>3</td></tr>
<tr><td colspan="5">Course Content</td><td>Hrs</td></tr>
<tr><td>Unit 1</td><td colspan="4">Introduction: Goals and history of IR. The impact of the web on IR.
Basic IR Models: Boolean and vector-space retrieval models; ranked retrieval; text-similarity metrics; TF-IDF (term frequency/inverse document frequency) weighting; cosine similarity.
Basic Tokenizing, Indexing, and Implementation of Vector-Space Retrieval:
Simple tokenizing, stop-word removal, and stemming; inverted indices; efficient processing with sparse vectors;</td><td>8</td></tr>
<tr><td>Unit 2</td><td colspan="4">Performance metrics: recall, precision, F-measure, and NDCG; Evaluations on benchmark text collections
Query Operations: Relevance feedback; Query expansion.
Text Representation: Word statistics; Zipf's law; Porter stemmer; morphology; index term selection; using thesauri.</td><td>8</td></tr>
<tr><td>Unit 3</td><td colspan="4">Web Search: Search engines; spidering; meta-crawlers; directed spidering; link analysis, HITS, hubs and authorities, Google PageRank);
Text Categorization: Categorization algorithms: Rocchio, nearest neighbor</td><td>8</td></tr>
<tr><td>Unit 4</td><td colspan="4">Text Classification: Language-Model Based Retrieval : Using naive Bayes text classification for ad hoc retrieval. Improved smoothing for document retrieval.
Text Clustering: Clustering algorithms: agglomerative clustering; k-means; expectation maximization (EM). Applications to web search and information organization.
Recommender Systems: Read this paper by Herlocker et al. Collaborative filtering and content-based recommendation of documents and products.</td><td>8</td></tr>
<tr><td>Unit 5</td><td colspan="4">Recommender Systems: Collaborative filtering and content-based recommendation of documents and products.
Ethical Issues in IR: Privacy, Fairness, Fake news and disinformation, Filter bubble, Viewpoint diversity, fostering extremism, Internet addiction.
Information Extraction and Integration: Extracting data from text; semantic web; collecting and integrating specialized information on the web.
Question Answering: Semantic parsing. Question Answering from structured data and text.
Deep Learning for IR: Word embeddings. Neural language models.</td><td>8</td></tr>
<tr><td colspan="6">References</td></tr>
<tr><td>1.</td><td colspan="5">Modern Information Retrieval, Ricardo Baeza-Yates and Berthier Ribeiro-Neto, Addison-Wesley, 2000. http://people.ischool.berkeley.edu/~hearst/irbook/</td></tr>
</table>

2.	Information Retrieval: Implementing and Evaluating Search Engines by S. Buttcher, C. Clarke and G. Cormack, MIT Press, 2010.
3.	Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data by B. Liu, Springer, Second Edition, 2011.
4.	Cross-Language Information Retrieval by By Jian-Yun Nie Morgan & Claypool Publisher series 2010
5.	Multimedia Information Retrieval by Stefan M. Rüger Morgan & Claypool Publisher series 2010
6	Ricci, F.; Rokach, L.; Shapira, B.; Kantor, P.B. (Eds.), Recommender Systems Handbook. 1st Edition., 2011, 845 p. 20 illus., Hardcover, ISBN: 978-0-387-85819-7 Relevant Research Papers

Data Mining and Warehousing					
Prerequisite: :		L	T	P	C
Total hours: 42		3	0	0	3
Course Content					Hrs
Unit 1	Overview of the Data Mining and Knowledge Discovery from Databases Process, Data Warehousing and OLAP, Data Preprocessing: Summary Data Structures, dimensionality reduction				8
Unit 2	Association Rule Mining: Frequent Item set Mining Methods, Rule Generation, Interestingness Measures				6
Unit 3	Classification: Decision Trees, Instance Based, Support Vector Machines, Computational Learning Theory, Associative Classification. Clustering: Partitional, Hierarchical, Density Based, Grid Based, Advanced Methods				10
Unit 4	Sequence Mining, Complex Data Mining. Web Mining: Information Retrieval, Link Analysis, Search Engines, Usage Analysis. Data Mining Applications				8
Unit 5	Data warehouse modelling: schema for multidimensional data models, concept hierarchies, Measures: categorization and computations.				8
References					
1.	1. J. Han and M. Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann/Elsevier India, 3rd edition, 2011.				
2.	Ian H. Witten and Eibe Frank, Data Mining: Practical Machine Learning Tools and Techniques (Second Edition), Morgan Kaufmann, 2005, ISBN: 0-12-088407-0.				

Digital Image Processing Lab				
Prerequisite: Fundamental knowledge on image processing and programming skills	L	T	P	C
	0	0	3	2

Course Content

1. Familiarization with various image processing tools
2. Basic operations on images
3. Basic grey-level transformations
4. Image Negative
5. Logarithmic transformation
6. Power-law transformation
7. Perform the following over a given image
8. Grey level slicing
9. Zooming (Nearest neighbour interpolation, bilinear interpolation)
10. Bit-plane slicing
11. Histogram equalization and specification
12. Implementation of different image transforms (DFT, DCT, DWT, etc.)
13. Spatial filtering in presence of various noise
14. Filtering in frequency domain
15. Implementation of image deblurring techniques
16. Image segmentation (edge detection, line detection, point detection)
17. Implementation of region based image segmentation
18. Implementation of different morphological operations
19. Analysis of images using color models
20. Mini project

References

1.	Rafael C. Gonzalez, Richard E. Woods, Digital Image Processing, Pearson , 3rd Edition, 2008
3.	Anil K. Jain, Fundamentals of Digital Image Processing, Pearson , 2002

Machine Learning Lab					
Prerequisite: : Python Programming		L	T	P	C
Total hours: 42		0	0	4	2
Course Content					Hrs
1	Perceptron learning algorithm for linear classifiocation.				3
2	**Linear Regression: using closed form solution and gradient descent**				3
3	**Logistic Regression for classification and linearly and non linearly separable data.**e				6
4	**Neural Networks:** In this problem you will implement forward and backward propagation methods for a multi-layer neural network with K hidden layers.				6
5	**Evaluation Metrics:** Taking two Gaussian distribution for genuine and imposter scores for a biometrics matcher. Plot the DET, ROC and AUC curve of this matcher. Also compute TPR, FPR, FRR, FAR, Specificity, Sensitivity, F1 score, Precison and Recall.				4
6	**SVM:** Classify the digits data as given for exercise 4 using a Support Vector Machine. Compute the values of W and an offset b, also draw the hyperplane.				8
7	**Decision Trees and Random Forest:** Generate 100 samples randomly for any classification problem with k attributes and n samples per bootstrap. And implement decision trees, bagging, boosting and random forest on this data.				6
8	**Clustering: I**mplement Agglomerative, Hierarchichal and Density based clustering techniques				6
References					
A first course in Machine learning, Simon Rogers and mark Girolami, CRC Press					
Learning from Data, Yaser S Abu-Mostafa, AML books					
Machine learning, Marsland, CRC press					
An Introduction to Machine Learning, Kubat Miroslav, Springer					

Big Data Analytics Lab				
Prerequisite: Nil	L	T	P	C
Total hours: 35	0	0	2	1

Course Content		Hrs
1.	Visualization: a. Find the data distributions using box and scatter plot. b. Find the outliers using plot. c. Plot the histogram, bar chart and pie chart on sample data	6
2.	R as Calculator Applications a. Using with and without R objects on console b. Using mathematical functions on console c. Write an R script, to create R objects for calculator application and save in a specified location in disk	6
3.	Descriptive statistics in r a. Write an R script to find basic descriptive statistics using summary b. Write an R script to find subset of dataset by using subset ()	9
4.	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.	3
5.	Apply multiple regressions, if data have a continuous independent variable. Apply on above dataset.	6
6.	a. Install relevant package for classification. b. Choose classifier for classification problem. c. Evaluate the performance of classifier.	5
7.	Installing Hadoop, PIG, Hive, Visualizing Big data sets, Applying Parallel machine learning models to handle large scale data.	

References	
1.	Wes McKinney, Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython, O'Reilly Media, 2017
2.	Joshua N. Milligan, Learning Tableau 2020: Create effective data visualizations, build interactive visual analytics and transform your organization, Packt Publishing Limited, 2020.
3.	Nathan Marz, James Warren: Big Data: Principles and best practices of scalable realtime data systems, 2020.
4.	Wes McKinney, Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython, O'Reilly Media, 2017

Bio-Medical Image Analysis				
Prerequisite: Knowledge of image processing and machine learning, basics of linear algebra and calculus, and programming skills	L	T	P	C
Total hours: 42	3	0	0	3

Course Content		Hrs.
Unit 1	Introduction to medical imaging, survey of major imaging modalities used for medical imaging: ultrasound, X-ray, CT, MRI, PET, and SPECT, surgical applications of medical image processing.	6
Unit 2	Image Enhancement and Image Segmentation: data acquisition, filtering, image enhancement, convolution, Image Transforms, interpolation, noise reduction methods, edge detection, Image segmentation	8
Unit 3	Basic concepts and algorithms in machine learning and deep learning: feature extraction and selection, neural networks and support vector machine, auto-encoder and its variants, convolutional neural networks (CNNs), transfer learning.	8
Unit 4	Computer aided diagnosis (CAD), History and success stories, Machine learning based CAD, CAD for various medical image analysis tasks, Case studies on recent advances in analysis of retinal fundus, CT, MRI, X-ray ultrasound and histopathology images	10
Unit 5	Medical image segmentation and registration Deep learning for medical image analysis: 3D Convolutional Neural Networks, Generative models for synthetic image generation	10

References	
1.	Anke Meyer-Baese, and Volker J. Schmid. Pattern recognition and signal analysis in medical imaging, Academic Press, 2014.
2.	Zhou, Kevin, Hayit Greenspan, and Dinggang Shen. Deep learning for Medical Image Analysis, Academic Press, 2017.
3.	Rafael C. Gonzalez and Richard E. Woods, Digital Image Processing, Pearson Education, 3rd Ed, 2009.
4.	Ian Goodfellow, Yoshua Bengio, Aaron Courville, Deep Learning, 2016

Social Network Analysis					
Prerequisite: Knowledge of computer networks		L	T	P	C
Total hours: 40		3	0	0	3
Course Content					Hrs
Unit 1	Social Network Analysis: Preliminaries and definitions, Erdos Number Project, Centrality measures, Balance and Homophily. Random graph models: Random graphs and alternative models, Models of network growth, Navigation in social Networks				8
Unit 2	Network topology and diffusion, Contagion in Networks, Complex contagion, Percolation and information, Epidemics and information cascades, Cohesive subgroups, Multidimensional Scaling, Structural equivalence, Roles and positions, Ego networks, Weak ties, Structural holes				10
Unit 3	Small world experiments, Small world models, Origins of small world, Heavy tails, Small Diameter, Clustering of connectivity, The Erdos Renyi Model, Clustering Models, Preferential Attachment				10
Unit 4	Navigation in Networks Revisited, Important vertices and page rank algorithm, Towards rational dynamics in networks, Basics of game theory				6
Unit 5	Coloring and consensus, biased voting, network formation games, network structure and equilibrium, behavioral experiments, Spatial and agent-based models				6

References	
1.	Wasserman, Stanley, and Joseph Galaskiewicz. Advances in social network analysis: Research in the social and behavioral sciences. Sage, 1994
2.	Knoke, David, and Song Yang. Social network analysis. Sage Publications, 2019
3.	Carrington, Peter J., John Scott, and Stanley Wasserman, eds. Models and methods in social network analysis. Vol. 28. Cambridge university press, 2005
4.	Liu, Bing. "Social network analysis." In Web data mining, pp. 269- 309. Springer, Berlin, Heidelberg, 2011

Sixth Semester					
S. No	**Code**	**Subject**	**L-T-P**	**Credits**	**Type**
	22AIT3xx	Deep Learning	3-0-0	3	PC
	22AIT3xx	Natural Language Processing	3-0-0	3	PC
	22AIT3xx	High Performance Computing	3-0-0	3	PC
	22AIT3xx	Information Security	3-0-0	3	PC
	22AIT3xx	Program Elective-2	3-0-0	3	PE
	22ECxxx	Wireless and 5G Communication	3-0-0	3	PLEAS
	22AIP3xx	Deep Learning Lab	0-0-4	2	PC
	22AIP3xx	Natural Language Processing Lab	0-0-2	1	PC
	22AIP3xx	High Performance Computing Lab	0-0-2	1	PC
			27	**22**	

Honors					
	22AITxxx	Honors Elective-1		3	
	22AITxxx	Honors Elective-2		3	
				6	

Minor AIDE					
	22AIT2xx	Database Management Systems	3-0-0	3	
	22AIT2xx	Artificial Intelligence	3-0-0	3	

Deep Learning					
Prerequisite: : Probability, Statistics, Algebra, Basic Computer Programming, Data Structures		L	T	P	C
Total hours: 42		3	0	0	3
Course Content					Hrs
Unit 1	Course Overview: Introduction to Deep Learning and its Applications. Introduction to Statistical Learning: Multi-Layer Perceptron, Back Propagation, Linear Regression, Loss Functions and Optimization: Optimization, stochastic gradient descent, dropout, batch normalization, etc.				8
Unit 2	Convolutional Neural Networks: Convolution, pooling, Activation Functions, Back propagation of CNN, Weights as templates, Translation invariance, Training with shared parameters. CNN Architecture Design and Discussion: AlexNet, VGG, GoogLeNet, ResNet, Capsule Net, etc. Visualization and Understanding: Visualizing intermediate features and outputs, Saliency maps, Visualizing neurons, Cam-Grad, etc.				8
Unit 3	Sequential Modelling: Recurrent and Recursive Nets, RNN, LSTM, GRU, Image captioning, visual question answering, etc.				6
Unit 4	Generative Models: Encoder, Decoders, Variational Autoencoders, Generative Adversarial Networks like pix2pix, CycleGAN, etc. Transformers based Models				8
Unit 5	Deep Learning Applications: Object Detection: RCNN, Fast RCNN, Faster RCNN, YOLO and variants, Retina Net, etc., Adversarial Attacks on CNN Deep learning for NLP				8
Unit 6	Deep learning Libraries and Frameworks: Keras, TensorFlow, PyTorch, AutoML, etc				4
References					
1.	Ian Goodfellow and Yoshua Bengio and Aaron Courville, “Deep Learning,” MIT Press.				
2.	Michael A. Nielsen, “Neural Networks and Deep Learning,” Determination Press, 2015.				

Natural Language Processing					
Prerequisite:		L	T	P	C
Total hours: 42		3	0	0	3
Course Content					Hrs
Unit 1	Introduction to NLP - Various stages of NLP –The Ambiguity of Language: Why NLP Is DifficultParts of Speech: Nouns and Pronouns, Words: Determiners and adjectives, verbs, Phrase Structure. Statistics Essential Information Theory : Entropy, perplexity, The relation to language, Cross Entropy, Character Encoding, Word Segmentation, Sentence Segmentation, Introduction to Corpora, Corpora Analysis. Inflectional and Derivation Morphology, Morphological analysis and generation using Finite State Automata and Finite State transducer.				6
Unit 2	Language Modelling, Words: Collocations- Frequency-Mean and Variance – Hypothesis testing:The t test, Hypothesis testing of differences, Pearson's chi-square test, Likelihood ratios. Statistical Inference: n –gram Models over Sparse Data: Bins: Forming Equivalence Classes- N gram model – Statistical Estimators-Combining Estimators				6
Unit 3	Word Sense Disambiguation, Methodological Preliminaries, Supervised Disambiguation: Bayesian classification, An informationtheoretic approach, Dictionary-Based Disambiguation: Disambiguation based on sense, Thesaurusbased disambiguation, Disambiguation based on translations in a second-language corpus.				6
Unit 4	Markov Model: Hidden Markov model, Fundamentals, Probability of properties, Parameter estimation, Variants, Multiple input observation. The Information Sources in Tagging: Markov model taggers, Viterbi algorithm, Applying HMMs to POS tagging, Applications of Tagging				6
Unit 5	Parsing, The Probability of a String, Problems with the Inside-Outside Algorithm, Parsing for disambiguation, Treebanks, Parsing models vs. language models, Phrase structure grammars and dependency, Lexicalized models using derivational histories, Dependency-based models.				8
Unit 6	Shallow Parsing and Chunking, Shallow Parsing with Conditional Random Fields (CRF), Lexical Semantics, WordNet, Thematic Roles, Semantic Role Labelling with CRFs. Statistical Alignment and Machine Translation, Text alignment, Word alignment, Information extraction, Text mining, Information Retrieval, NL interfaces, Sentimental Analysis, Question Answering Systems, Social network analysis, Text Summarization. Introduction to LLMs (Large Langauge Models)				10
References					
1.	D. Jurafsky, J.H. Martin, Speech and Language Processing, 3rd Online Edition (available at https://web.stanford.edu/~jurafsky/slp3/).				
2.	J. Eisenstein, Introduction to Natural Language Processing, MIT Press, 2019.				

High Performance Computing					
Pre-requisites: Data Structures / Operating Systems / Computer Networks		L	T	P	C
Total Hours: 42		3	0	0	3
Course Contents				Hrs	
Unit 1	Sequential Programming, Concurrent Programming and Parallel Programming; Parallel Programming Paradigms – Data Parallel, Task Parallel, Shared Memory, Message Passing; Co-Processors and Accelerators; Parallel Programs, Performance Metrics for Parallel Systems (Speed Up, Performance, Cost, and Scalability analysis), Amdahl's Law, Gustafson Law; Multi-core Architecture: An overview of Parallel Computing Platforms, Flynn's Classification, Single-Core and Multi-Processor, General Purpose Graphical Processing Unit (GPGPU).			**6**	
Unit 2	Programming Shared Address Space Platforms: • OpenMP – A standard for Directive Parallel Programming; The OpenMP programming Model (Concurrent Tasks, Synchronization Constructs, Data Handling); OpenMP-Environment Variables. • POSIX threads (Pthreads), Synchronization primitives, Threads- mutex and condition variables, Synchronization constructs, Threads Versus OpenMP • Intel Threading Building Blocks (TBB) – Express Parallelism in C++ program; Containers, Scalable Memory Allocation; Mutual Exclusion; Task Scheduler			**8**	
Unit 3	Programming Using the Message-Passing Paradigm: Message Passing Interface (MPI): Principles of Message Passing Programming; MPI Building blocks (Send and Receive Operations); Blocking & Non – Blocking Communication; Collective Communication and Computation Operations;			**8**	
Unit 4	Parallel Programming • Matrix Computations – Matrix-Vector Multiplication, Matrix-Matrix Multiplication; Sparse Matrix Computations with Vector • Sorting algorithms (Bubble Sort and Quicksort); • Sequential & Parallel Search Algorithms; Depth-First Search Algorithms; Best-First Search Algorithms; • Graph Algorithms: All-pairs of Shortest Paths Algorithms;			**10**	
Unit 5	Programming on Multi-Core Systems with GPU accelerators: • History of GPUs; GPGPU Programming; GPU Memory Hierarchy Features; An Overview of CUDA enabled NVIDIA GPUs, Introduction to CUDA C, • The OpenCL – Heterogeneous Programming; OpenCL Libraries, The OpenCL Memory Model, Execution Model; Platform and Devices; An Overview of OpenCL API; Python-GPU (PyCUDA, NumPy)			**10**	

References

1	A. Grama, A. Gupta, G. Karypis, and V. Kumar, *Introduction to Parallel Computing*, 2nd Ed, Pearson Education, 2007, ISBN: 978-0201648652
2	P. Pacheco, *An Introduction to Parallel Programming*, 1st Ed, Morgan Kaufman, 2011, ISBN: 978-0123742605
3	B. Chapman, G. Jost and R. Pas, *Using OpenMP: Portable Shared Memory Parallel Programming*, The MIT Press, 2008, ISBN: 978-0262533027
4	R. Chandra, R. Menon, L. Dagum, D. Kohr, D. Maydan, and J. McDonald, *Parallel Programming in OpenMP*, 1st Ed, Morgan Kaufmann, 2000, ISBN: 978-1558606715
5	R. Pas, E. Stotzer, and C. Terboven, *Using OpenMP-The Next Step Affinity – Accelerators, Tasking, and SIMD*, The MIT Press, 2017, ISBN: 978-0262534789

6	M. Snir, S. Otto, S. Huss-Lederman, D. Walker, and J. Dongarra, *MPI – The Complete Reference, Vol 1 – The MPI Core*, 2nd Ed, The MIT Press, 1998, ISBN: 78-0262692151
7	W. Gropp, E. Lusk, N. Doss, A. Skjellum, Using MPI: Portable Parallel Programming with the Message Passing Interface, 3rd Ed, The MIT Press, 1994, ISBN: 978-026252739
8	M. J. Quinn, *Parallel Programming in C with MPI and OpenMP*, 1st Ed, McGraw Hill, 2017, ISBN: 978-0070582019
9	J. Sanders and E. Kandrot, *CUDA By Example – An Introduction to General-Purpose GPU Programming*, 1st Ed, Addison Wesley, 2011, ISBN: 978-0131387683
10	D. B. Kirk and W. W. Hwu, *Programming Massively Parallel Processors: A Hands-on Approach Paperback*, 1st Ed, Morgan Kaufmann, 2010, ISBN: 978-0123814722
11	D. Kaeli, P. Mistry, D. Schaa, D. P. Zhang, *Heterogeneous Computing with OpenCL 2.0*, 3rd Ed, Morgan Kaufmann, 2015, ISBN: 978-0128014141

Information Security					
Prerequisite: : Cryptography, Computer networks, etc.		L	T	P	C
Total hours: 42		3	0	0	3
Course Content					Hrs
Unit - I	**Introduction to Resource Protection** - Resource protection models; - Introduction to Cryptography Concepts - Private Key & Public Key Cryptography, Hash Functions, MACs, Digital Signatures, Authentication, Key Establishment, Kerberos, DH key exchange - Man-in-the-middle, Freshness				10
Unit - II	**Software and OS Security:** OS Security: Common Bugs, Buffer Overflow, Runtime Defenses against memory safety vulnerabilities, program verification and other vulnerabilities, Principles in OS Security; Mechanisms for confining bad code, Mechanisms for confining bad code: isolation, sandboxing, SFI and Virtualization, Trusted Computing				7
Unit - III	**Trust Computing and Data Security: -** Trust, Trusted Computing, Hardware-assisted Security, Applications of Trusted Computing**,** Secure web site design (SQL injection, XSS, etc.), Browser Security,				7
Unit - IV	**Data Security and integrity - 2:** PKI, CAs, TLS, IPSec, Sniffing, Spoofing, Enumeration, Nmap - Vulnerability Scanning, Metasploit, DDoS, Worms, Viruses, Trojans, Botnets, DNS , CDNs, Firewalls, Spam Filters, Deep-packet Inspection, Log-file Analysis, Intrusion Detection Systems, types of IDS, Log Aggregation, SIEM				11
Unit - V	**Data Privacy -** Anonymization models: K-anonymity, l-diversity, t-closeness, differential privacy, Statistical Database security Inference Control Secure Multi-party computation, Emerging Applications Social Network Privacy - Location Privacy, Query Log Privacy, Biomedical Privacy.				7
References					

1.	Security in Computing (3rd edition)
2.	Security Engineering, Second Edition, Ross Anderson. Wiley, 2008.
3.	Foundations of Security: What Every Programmer Needs to Know, Neil Daswani, Christoph Kern, Anita Kesavan. Apress, 2007.
4.	The Algorithmic Foundations of Differential Privacy, Cynthia Dwork and Aaron Roth
5.	Cryptography and Networks, William Stallings, 7 edition
6.	The course materials are mainly from the lecturing slides I?ve made and research papers from top conferences like NDSS, USENIX, SIGCOMM, MOBICOM, NSDI, MobiSys etc.

Wireless and 5G Communication					
Prerequisite: Computer networks		L	T	P	C
Total hours: 42		3	0	0	3
Course Content					Hrs
Unit - I	**Introduction to 5G:** Fundamentals of Wireless Communication, Evolution from 1G to 5G, 5G spectrum, Wireless Standards: Overview of 2G 3G, 4G and 5G, Key capabilities of 5G, System Architecture, Performance measures- Outage, average snr, average symbol/bit error rate. System examples- GSM, EDGE, GPRS, IS-95, CDMA 2000 and WCDMA, 3G, 4G and 5G mobile communications.				
Unit - II	**Cellular System Design Fundamentals:** Components of Mobile Cellular Systems: Cell structure, frequency reuse, cell splitting, Call origination & Termination. Cellular concepts- Signal propagation- Propagation mechanism, reflection, refraction, diffraction and scattering, large scale signal propagation and lognormal shadowing. Interference & System Capacity: Improving Capacity in Cellular Systems, Co-Channel Interference, Channel Assignment Strategies, Handoff Strategies.				
Unit - III	**Channel Fading and Diversity:** Multipath Measurements, Parameters of Mobile Multipath Channels, Types of Fading: Multipath and small-scale fading- Doppler shift, power delay profile, average and rms delay spread, coherence bandwidth and coherence time, flat and frequency selective fading, slow and fast fading, average fade duration and level crossing rate. Impulse Response Model of a Multipath Channel, Channel State Information. Receiver structure- Diversity receivers-selection and MRC receivers, RAKE receiver, equalization: linear-ZFE and adaptive, DFE. Transmit diversity Altamonte scheme				
Unit - IV	**5G Radio Standard:** Orthogonal frequency division multiplexing (OFDM), Modulation schemes- BPSK, QPSK and variants, QAM, MSK and GMSK, multicarrier modulation, OFDM, MIMO and space time signal processing, spatial multiplexing, diversity/multiplexing tradeoff.				
Unit - V	**5G Enabling Technologies:** Concept of 5G Communication, Multi-carrier with filtering, Filter-bank based multi-carrier, Non-orthogonal multiple access (NOMA). Principle and Spectrum Allocation, Power Control Mechanism in NOMA Techniques, 5G Applications.				

References	
1.	Wireless Communications: Principles & Practices by Theodore S. Rapport.
2.	Mobile Cellular Telecomm. B y William C. Y. Lee.
3.	Mobile Communication by Schiller, Pearson Education India.
4.	Osseiran, Afif, Jose F. Monserrat, and Patrick Marsch, eds. 5G mobile and wireless communications technology. Cambridge University Press, 2016.
5.	Rodriguez, Jonathan. Fundamentals of 5G mobile networks. John Wiley & Sons, 2015.

Deep Learning Lab					
Prerequisite: The programming lab in C++, which means you need to be very comfortable with C++ and using standard debugging tools.		L	T	P	C
Total hours: 36		0	0	4	2
Course Content					Hrs
	The Lab experiements may changed based on the course requirments. 1. Familiarization of cloud based computing like Google colab, GPU Programming 2. Basic image processing operations: Histogram equalization, thresholding, edge detection, data augmentation, morphological operations 3. Implement SVM/Softmax classifier for CIFAR-10 dataset: (i) using KNN, (ii) using 3 layer neural network 4. Study the effect of batch normalization and dropout in neural network classifier 5. Familiarization of image labelling tools for object detection, segmentation 6. Image segmentation using Mask RCNN, UNet, SegNet 7. Object detection with single-stage and two-stage detectors (Yolo, SSD, FRCNN, etc.) 8. Image Captioning with Vanilla RNNs 9. Image Captioning with LSTMs 10. Network Visualization: Saliency maps, Class Visualization 11. Generative Adversarial Networks 12. Chatbot using bi-directional LSTMs				36
References					
1.	Francois Chollet, “Deep learning with Python” – Manning Publications.				
2.	Michael A. Nielsen, “Neural Networks and Deep Learning,” Determination Press, 2015.				

Natural Language Processing Lab					
Prerequisite:		L	T	P	C
Total hours: 28		0	0	3	2
Course Content					Hrs
Unit 1	Implementation of Pre-processing of Text (Tokenization, Stop word Removal, Stemming and Lemmatization etc.) and Morphological Analysis				6
Unit 2	Implementation of N-gram Models				4
Unit 3	Implementation of Word Sense Disambiguation				2
Unit 4	Implementation of POS Tagging and Named Entity Recognition				6
Unit 5	Implementation of CKY Parsing and Mini Project, and basic LLMs Assignments				10
References					
1.	D. Jurafsky, J.H. Martin, Speech and Language Processing, 3rd Online Edition (available at https://web.stanford.edu/~jurafsky/slp3/).				
2.	J. Eisenstein, Introduction to Natural Language Processing, MIT Press, 2019.				

High Performance Computing lab				
Prerequisite: Basic Programming Skills.	L	T	P	C
Total hours: 36	0	0	2	1
Course Content				Hrs

The Lab experiments may change based on the course requirements.

MPI Assignments:

1) WAP for parallel BFS using MPI function call.
2) WAP for parallel DFS using MPI function call.
3) WAP for parallel search algorithm using MPI function call.
4) WAP for parallel binary search using MPI function call.
5) WAP for parallel bubble sort using MPI_send() and MPI_recv().
6) WAP for parallel quick sort using MPI function.
7) WAP for parallel sample sort algorithm using MPI function call.
8) WAP for parallel binary search using MPI function call.
9) Write MPI program to compute the vlaue of PI by numerical integration using MPI point-to-point blocking communication library calls.

Pthread Assignments:

1) Write a Pthread program to print *Hello World*
2) Write a Pthread program to find Sum of first *n* Natural Numbers
3) Write a Pthread program to illustrate pthread join operation
4) Write a Pthread program to illustrate basic stack management
5) Write a Pthread program to find the minimum of an array.
6) Write Pthread code to Find out minimum in an un-sorted integer array.
7) Write Pthread code to Find *k* matches in the list
8) Pthread prorgam to compute *Pie* value by Numerical Integration method
9) Write Pthread code to perform Vector-Vector Multiplication using block striped partitioning.
10) Write Pthread code to perform Vector-Vector Addition using block striped partitioning.
11) Write a Pthread program to find minimum value in an Integer array using *Mutex*.
12) Write a Pthread program to find minimum value in an Integer array using *Read Write Lock*.

OpenMP Assignments:

1) Write a OpenMP program to print unique identifier
2) Write a "Hello world" Program Using OpenMP pragmas
3) Illustrate a program for loop recurrence using OpenMP *PARALLEL FOR* directive
4) Write a OpenMP program to find Sum of Natural Numbers using OpenMP *Parallel FOR* directive
5) Write a OpenMP program to find Sum of Natural Numbers using OpenMP *REDUCTION* clause
6) Write a OpenMP program for Loop-carried dependence using OpenMP *parallel* Directive
7) Write a OpenMP program to illustrate *Data Race condition*
8) Write a OpenMP program to illustrate *Managing Shared & Private Data*
9) Write a OpenMP program to illustrate *Loop Scheduling & Partitioning - Optimize Performance*

Hrs: 36

	10) Write a OpenMP program to illustrate *Work-Sharing Sections* 11) Write a OpenMP program to illustrate the *performance improvement.* **CUDA Assignments:** 1) Write a CUDA program to compute Vector - Vector addition 2) Write a CUDA program to compute Matrix - Matrix addition 3) Write a CUDA Program to compute vector - Vector multiplication. 4) Write a CUDA Program to find prefix sum of a given array. 5) Write a CUDA program to find transpose of a matrix. 6) Write a CUDA Program to calculate value of PI using numerical integration method. 7) Write a CUDA Program to find infinity norm of a matrix. 8) Write a CUDA Program for Matrix Vector multiplication 9) Write a CUDA Program for Matrix Matrix multiplicationbased on tiling Partitioning
References	
1.	Chandra, Rohit. *Parallel programming in OpenMP*. Morgan kaufmann, 2001.
2	Pacheco, Peter, and Matthew Malensek. *An introduction to parallel programming*. Morgan Kaufmann, 2021.
3	Gropp, W., Lusk, E., & Skjellum, A. (1999). *Using MPI: portable parallel programming with the message-passing interface* (Vol. 1). MIT press.
4	Balaji, Pavan, ed. *Programming models for parallel computing*. MIT Press, 2015.
5	Soyata, T. (2018). *GPU parallel program development using CUDA*. CRC Press.
6	Cheng, J., Grossman, M., & McKercher, T. (2014). *Professional CUDA c programming*. John Wiley & Sons.

List of proposed Elective Courses

Area Cluster	***Specific Subject Titles***	***Course Credit***
Machine Learning & Intelligence Systems	1) Ethical AI	3-0-0
	2) IoT based Robotics#	*3-0-0*
	3) IoT based Robotics Labs#	*0-0-2*
	4) Game Theory and Strategic Decisions	3-0-0
	5) Cyber Physical Systems	3-0-0
	6) Natured Inspired Algorithms	3-0-0
	7) Optimization in ML	3-0-0
	8) Reinforcement learning	3-0-0
Data Analytics	*1) Data Visualization and Interpretation#*	*3-0-0*
	2) Data Visualization and Interpretation Lab#	*0-0-2*
	3) Time Series Analysis	3-0-0
	4) Graph Analytics	3-0-0
	5) Data Analytics	3-0-0
	6) Data Compression	3-0-0
High Performance Computing	1) Cloud Computing	3-0-0
	2) Quantum Computing	3-0-0
	3) Parallel and Distributed Computing#	*3-0-0*
	4) Parallel and Distributed Computing Lab#	*0-0-2*
	5) Parallelizing Compilers	3-0-0
	6) System on Chip	3-0-0
	7) Evolving Architectures#	*3-0-0*
	8) Evolving Architectures Lab#	*0-0-2*
	9) Distributed System	3-0-0
Speech, Vision, and Text	1) Deep Learning for NLP	3-0-0
	2) Computer Vision	*3-0-0*
	3) Computer Vision Lab	*0-0-2*
	4) Biometrics	*3-0-0*
	5) Biometrics Lab	*0-0-2*
	6) Large Language Models	3-0-0
Security	1) Cloud Security	3-0-0
	2) Digital Forensic	3-0-0
	3) Embedded System Security	3-0-0
	4) Intrusion Detection	3-0-0
	5) Cryptography	3-0-0
	6) Blockchain Technologies	3-0-0
General CS	*1) Advanced Algorithms#*	*3-0-0*
	2) Advanced Algorithms Lab#	*0-0-2*
	3) Advance compiler design#	*3-0-0*
	4) Advance compiler design Lab#	*0-0-2*
	5) Advance Database System #	*3-0-0*
	6) Advance Database System Lab#	*0-0-2*

	7) Software Testing and Validation	3-0-0
	8) Real Time System	3-0-0
	9) Wireless Sensor Networks	3-0-0
	10) Internet of Things	3-0-0
	11) Software Engineering	3-0-0
	12) Object Oriented Analysis and Design#	*3-0-0*
	13) Object Oriented Analysis and Design Lab#	*0-0-2*

 # Program Electives 5 and 7 and their respective labs must be chosen from these subjects only.

Ethical AI					
Prerequisite: Algorithms, AI, Social sciences		L	T	P	C
Total hours: 28		2	0	2	
Course Content					Hrs.
Unit 1	**Introduction:** Definition of morality and ethics in AI-Impact on society Impact on human psychology-Impact on the legal system-Impact on the environment and the planet-Impact on trust				7
Unit 2	Ethical initiatives in the field of AI: International ethical initiatives. Ethical harms and concerns tackled by these initiatives. Harms in detail, case studies.				7
Unit 3	**AI standards and regulations: National and international strategies in AI. Government readiness for AI.** Model Process for Addressing Ethical Concerns During System Design – Transparency of Autonomous Systems-Data Privacy Process-Algorithmic Bias Considerations Ontological Standard for Ethically Driven Robotics and Automation Systems				8
Unit 4	Emerging themes: Addressing ethical issues through national and international strategies. Addressing governance challenges posed by AI.				7
References					
1.	Y. Eleanor Bird, Jasmin Fox-Skelly, Nicola Jenner, Ruth Larbey, Emma Weitkamp and Alan Winfield, The ethics of artificial intelligence: Issues and initiatives, EPRS | European Parliamentary Research Service Scientific Foresight Unit (STOA) PE 634.452 – March 2020.				
2.	Patrick Lin, Keith Abney, George A Bekey, Robot Ethics: The Ethical and Social Implications of Robotics, The MIT Press- January 2014.				

IoT based Robotics					
Prerequisite: Nil		L	T	P	C
Total hours: 42		3	0	0	3
Course Content					Hrs
Unit 1	Introduction to IoT and Robotics: Overview of IoT and Robotics; Historical development of IoT and Robotics; Applications of IoT and Robotics; Types of IoT devices; Types of Robotics;				6
Unit 2	Introduction to the Internet of Things. Protocols and Architectures. IoT Hardware: IoT devices and sensors; IoT networks and communication protocols; IoT gateways and controllers; IoT platforms and services				10
Unit 3	IoT Software: Introduction to IoT protocols; IoT data management and analytics; IoT security and privacy; IoT programming and development;				8
Unit 4	Robotics Fundamentals: Robotics history and evolution; Robotics components and structure. Robotics Hardware: Types of robots and their applications; Robotics sensors and actuators; Robotics control systems; Robotics power systems. Robotics Software: Robotics programming and development; Robotics motion planning and control; Robotics perception and vision; Robotics intelligence and autonomy.				10
Unit 5	Robotics Applications: Industrial Robotics; Service Robotics; Medical Robotics				4
Unit 6	IoT and Robotics Integration: Use cases and examples; Challenges and opportunities; Future trends and directions				4

References

1.	The Internet of Things: Key Applications and Protocols, David Boswarthick, Olivier Hersent, and Omar Elloumi, Wiley
2.	Building the Internet of Things with IPv6 and MIPv6, Daniel Minoli, Wiley
3.	Learn Robotics Programming, Danny Staple, Packt Publishing, 2nd ed.
4.	Robotics Simplified, Jisu Elsa Jacob and Manjunath N, BPB Publications.

IoT based Robotics Lab					
Prerequisite:		L	T	P	C
Total hours: 28		0	0	3	2
Course Content					Hrs
Unit 1	Setting up communication using XBEE and BLE. Data Exchange and interfacing Sensors				8
Unit 2	Interfacing with Actuators. Programming Motion and automation. Controller based interfacing.				10
Unit 3	Visual interfacing and controlling motion. Analysis of robotic arm and conveyor belts.				10
References:					
1.	Learn Robotics Programming, Danny Staple, Packt Publishing, 2nd ed.				
2.	Robotics Simplified, Jisu Elsa Jacob and Manjunath N, BPB Publications.				

Game Theory and Strategic Decisions					
Prerequisite: Calculus, Linear Algebra,		L	T	P	C
Total hours: 40		3	0	0	3
Course Content					Hrs.
Unit 1	Introduction to Game Theory Basics of Game Theory, Types of Games (cooperative vs. non-cooperative, zero-sum vs. non-zero-sum), Key Concepts (players, strategies, payoffs), Applications in AI and decision-making				4
Unit 2	Basic Concepts in Non-Cooperative Games: Normal Form Games, Dominant Strategies and Dominance Solvability, Nash Equilibrium, Mixed Strategies and the concept of Utility, Prisoner's Dilemma and other classical examples				6
Unit 3	Extensive Form Games: Trees and Game Trees, Strategies in Extensive Form Games, Backward Induction, Subgame Perfect Equilibrium, Dynamic Games in Repeated Interactions				6
Unit 4	Cooperative Game Theory: Basics of Cooperative Game Theory, Characteristic Function Games, Shapley Value, Core and Imputations, Applications in resource allocation and coalition formation				4
Unit 5	Evolutionary Game Theory and Game Theory in AI and Multi-Agent Systems: Game-Theoretic Modeling of Multi-Agent Systems, Game-Theoretic Learning (e.g., fictitious play, reinforcement learning), Auctions and Mechanism Design, AI applications of Game Theory (e.g., adversarial settings, negotiation)				10
Unit 6	Advanced Topics and Applications: Bayesian Games, Behavioral Game Theory, Game Theory in Social Networks, Ethical and strategic decision-making in AI, Analyzing real-world case studies involving game theory				9
References					
1.	"Game Theory" by Drew Fudenberg and Jean Tirole				
2.	"Algorithmic Game Theory" by Noam Nisan, Tim Roughgarden, Eva Tardos, and Vijay V. Vazirani				
3.	Evolutionary Games and Population Dynamics" by Josef Hofbauer and Karl Sigmund				
4.	"Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations" by Yoav Shoham and Kevin Leyton-Brown				

Cyber-Physical Systems (CPS)					
Prerequisite: Data communication, Computer networks, etc.		L	T	P	C
Total hours: 40		3	0	0	3
Course Content					Hrs
Unit - I	**Introduction to CPS:** Characteristics of Cyber-Physical Systems (CPS), Cyber-Physical Systems (CPS) in the real world, Basic principles of design and validation of,				4
Unit - II	**CPS Hardware:** Industry 4.0, AutoSAR, IIOT implications, Building Automation, Medical CPS -CPS physical systems modeling and formalisms: CPS - Platform components - CPS HW platforms - Processors, Sensors, Actuators,				8
Unit - III	**CPS Network and systems:** CPS Network - WirelessHart, CAN, Automotive Ethernet, Scheduling Real Time CPS tasks Principles of Dynamical Systems - Dynamical Systems and Stability, Controller Design Techniques and Performance under Packet drop and Noise,				11
Unit - IV	**CPS Implementations and Intelligence:** CPS implementation issues - From features to automotive software components, Mapping software components to ECUs, CPS Performance Analysis - effect of scheduling, bus latency, sense and actuation faults on control performance, network congestion, and building real-time networks for CPS, CPS Intelligent CPS				11
Unit - V	**Applications and Security for CPS:** Safe Reinforcement Learning, Robot motion control, Autonomous Vehicle control, Gaussian Process Learning, Smart Grid Demand Response, Building Automation, Secure Deployment of CPS, Secure Task mapping and Partitioning, State estimation for attack detection, Automotive Case study: Vehicle ABS hacking Power Distribution Case study: Attacks on Smart Grids				6

References	
1.	"Introduction to Embedded Systems – A Cyber–Physical Systems Approach" - E. A. Lee, Sanjit Seshia
2.	"Principles of Cyber-Physical Systems" - Rajeev Alur
3.	Research papers from top conferences like SIGCOMM, MOBICOM, NSDI, MobiSys etc.

Nature Inspired Algorithms					
Prerequisite: **Programming in C**		L	T	P	C
Total hours: 40		3	0	0	3
Course Content					Hrs
Unit 1	Introduction to Algorithms, Optimization, and Search for optimality, computational intelligence, Nature Inspired solutions and characteristic, Nature inspired Meta-heuristics and its brief history.				8
Unit 2	Analysis of Optimization Algorithms, Nature Inspired Algorithms, parameter Tuning and control Constrained and unconstrained optimizations, Random Walks and Optimizations, evolutionary strategies and Evolutionary Algorithms (EA), Simulated Annealing (SA) Algorithm and its behaviour, Genetic Algorithms (GA)- genetic operator, parameters, fitness functions, genetic programming and convergence analysis, GA variants				10
Unit 3	Swarm Intelligence optimization, Particle Swarm Optimization (PSO) Algorithm, Ant Colony Optimization (ACO) Algorithms, Artificial Bee Colony ACO) optimization algorithms, Cuckoo Search (CS) Algorithms, Intelligent Water Drop Algorithm (IWD), Bat Algorithms (BA), Firefly Algorithms (FA)				8
Unit 4	Applications of nature-inspired algorithm, machine learning using nature inspired algorithm, data clustering using NIA.				6
Unit 5	Parallel processing of NIA using Hadoop, Parallel data clustering using NIA. Multi-objective optimization and applications.				8

References	
1.	Nature-Inspired Optimization Algorithms – by Xin-She Yang (Author), June 30, 2016
2.	Mathematical Foundations of Nature-Inspired Algorithms, Xin-She Yang, Xing-Shi He, Springer; 1st ed. 2019 edition
3.	Evolutionary Algorithms in Engineering Applications, Editors: Dipankar Dasgupta and Zbigniew Michalewicz, Springer-Verlag, 1997
4.	Introduction to Evolutionary Computing, A. E Eiben and J. E. Smith, Second Printing, Springer, 2007

Optimization in ML					
Prerequisite: Calculus, Linear Algebra, Basics of ML and DL		L	T	P	C
Total hours: 40		3	0	0	3
Course Content					Hrs.
Unit 1	Introduction to Optimization in ML: Overview of optimization in machine learning, Types of optimization problems in ML, Importance of optimization in ML				3
Unit 2	Convex Optimization: Convex sets and functions convex optimization problems, Convex optimization algorithms (e.g., Gradient Descent, Newton's method), Convexity in ML models (e.g., linear regression)				6
Unit 3	Non-convex Optimization: Challenges in non-convex optimization, Gradient-based optimization (e.g., Stochastic Gradient Descent), Second-order optimization (e.g., L-BFGS), Handling constraints in non-convex optimization				8
Unit 4	Regularization and Optimization: Regularization techniques (L1, L2, Elastic Net), Regularization as a form of optimization, Role of regularization in preventing overfitting.				4
Unit 5	Optimization for Deep Learning: Optimization challenges in deep neural networks, Adaptive learning rate methods (e.g., Adam, RMSProp), Batch normalization and optimization, Optimization for recurrent neural networks, Weight initialization techniques				10
Unit 6	Hyperparameter Tuning and Advanced topics: Importance of hyperparameters in ML models, Grid search and random search for hyperparameter tuning, Bayesian optimization for hyperparameter tuning, Optimization for reinforcement learning, Optimization for generative models (e.g., GANs), Optimization in online and distributed learning				9

References

1.	"Convex Optimization" by Stephen Boyd and Lieven Vandenberghe
2.	" Machine Learning: A Probabilistic Perspective" by Kevin P. Murphy
3.	"Optimization Methods in Machine Learning" by Léon Bottou, Frank E. Curtis, and Jorge Nocedal
4.	"Numerical Optimization" by Jorge Nocedal and Stephen J. Wright

Reinforcement Learning					
Prerequisite:		L	T	P	C
Total hours: 40		3	0	0	3
Course Content					Hrs.
Unit 1	Basics of probability and linear algebra, Definition of a stochastic multi-armed bandit, Definition of regret, Achieving sublinear regret, UCB algorithm, KLUCB, Thompson Sampling				7
Unit 2	Markov Decision Problem, policy, and value function, Reward models (infinite discounted, total, finite horizon, and average), Episodic & continuing tasks, Bellman's optimality operator, and Value iteration & policy iteration				7
Unit 3	The Reinforcement Learning problem, prediction and control problems, Model based algorithm, Monte Carlo methods for prediction, and Online implementation of Monte Carlo policy evaluation				7
Unit 4	Bootstrapping; TD(0) algorithm; Convergence of Monte Carlo and batch TD(0) algorithms; Model-free control: Q-learning, Sarsa, Expected Sarsa.				7
Unit 5	n-step returns; TD(λ) algorithm; Need for generalization in practice; Linear function approximation and geometric view; Linear TD(λ).				6
Unit 6	Tile coding; Control with function approximation; Policy search; Policy gradient methods; Experience replay; Fitted Q Iteration; Case studies				6
References					
1.	Sutton, Richard S., and Andrew G. Barto. "Reinforcement learning: An introduction," First Edition, MIT press				
2.	Sugiyama, Masashi. "Statistical reinforcement learning: modern machine learning approaches," First Edition, CRC Press				
3.	Lattimore, T. and C. Szepesvári. "Bandit algorithms," First Edition, Cambridge University Press				
4.	Boris Belousov, Hany Abdulsamad, Pascal Klink, Simone Parisi, and Jan Peters "Reinforcement Learning Algorithms: Analysis and Applications," First Edition, Springer				

Data Visualisation and Interpretation					
Prerequisite:		L	T	P	C
Total hours: 30		3	0	0	3
Course Content					Hrs.
Unit 1	Data visualization and Interpretation-Visualization as a Discovery tool, Visualization skills for the masses, The Visualization methodology, Visualization design objectives, Exploratory vs. explanatory analysis, Understanding the context for data presentations, 3 minutes story, Effective Visuals, Gestalt principles of visual perception, Visual Ordering, Decluttering, Story Telling, Visualization Design;				10
Unit 2	Taxonomy of Data Visualization Methods: Exploring Tableau, Dashboard and Stories, Bullet graphs, Pareto charts, Custom background images;				10
Unit 3	Dashboard: Dashboard categorization and typical data, Characteristics of a Well-Designed Dashboard, Key Goals in the Visual Design Process; Power of Visual Perception: Visually Encoding Data for Rapid Perception, Applying the Principles of Visual Perception to Dashboard Design				10
Unit 4	Visualization using Matplotlib, Seaborn, Bokeh				6

References

1.	Knaflic, Cole. Storytelling With Data: A Data Visualization Guide for Business Professionals, Wiley
2.	Post, Frits H., Gregory Nielson, and Georges-Pierre Bonneau, eds. "Data visualization: The state of the art." (2002).
3.	Healy, K. (2018). Data visualization: a practical introduction. Princeton University Press.
4	Steve and Jefffrey, The big book of dashboards: visualizing your data using real world business scenarios. Wiley

Data Visualization and Interpretation lab					
Prerequisite: Basic Programming Skills.		L	T	P	C
Total hours: 36		0	0	2	1
Course Content					Hrs
	The Lab experiments may changed based on the course requirements. 1. Introduction to various Data Visualization tools 2. Basic Visualization in Python 3. Implementation of Visualization packages such as seaborn, matplotlib etc 4. Dashboard in Python 5. Introduction to Tableau and Installation 6. Connecting to Data and preparing data for visualization in Tableau 7. Data Aggregation and Statistical functions in Tableau 8. Data Visualizations in Tableau 8. Basic Dashboards in Tableau 9. Measure of Dispersion (Grouped Data). 10. Moment, Measures of Skewness & Kurtosis (Ungrouped Data). 11. Moments, Measures of Skewness & Kurtosis (Grouped Data). 12. Correlation & Regression Analysis. 13. Application of One Sample t – test. 14. Application of Two Sample Fisher’s t – test. 15. Chi – Square test of Goodness of Fit. 16. Chi – Square test of independent of Attributes for 2 X 2 contingency table. 17. Analysis of Variance One Way Classification. 18. Analysis of Variance Two Way Classification. 19. Selection of Random Sample Using Simple Random Sampling.				36
References					
1.	Data visualization with python: create an impact with meaningful data insights using interactive and engaging visuals, Mario Dobler, Tim Grobmann, Packt Publications, 2019				
2.	Practical Tableau: 100 Tips, Tutorials, and Strategies from a Tableau Zen Master, Ryan Sleeper, Oreilly Publications, 2018				

Time Series Analysis				
Prerequisite: Nil	L	T	P	C
Total hours: 40	3	0	0	3

	Course Content	Hrs.
Unit 1	Basic Properties of time-series data: Distribution and moments, Stationarity, Autocorrelation, Heteroscedasticity, Normality	8
Unit 2	Autoregressive models and forecasting: AR, ARMA, ARIMA models, Random walk model: non-stationarity and unit-root process, Drift and Trend models	8
Unit 3	Regression analysis with time-series data, Principal Component Analysis (PCA) and Factor Analysis	7
Unit 4	Conditional Heteroscedastic Models: ARCH, GARCH. T-GARCH, BEKKGARCH, Introduction to Non-linear and regime-switching models: Markov regime-switching models, Quantile regression, Contagion models	9
Unit 5	Introduction to Vector Auto-regressive (VAR) models: Impulse Response Function (IRF), Error Correction Models, Co-integration, Introduction to Panel data models: Fixed-Effect and Random-Effect models	8

References

1.	Ruey S. Tsay "Analysis of Time-series data," Third Edition, Wiley, 2014
2.	Chris Brooks "Introductory Econometrics for Finance," Fourth Edition, Cambridge University Press, 2019
3.	John Fox and Sanford Weisberg "An R Companion to Applied Regression," Third Edition, SAGE, 2018
4.	Yves Croissant and Giovanni Millo "Panel Data Econometrics with R," First Edition, Wiley, 2018

Graph Analytics					
Prerequisite: Nil		L	T	P	C
Total hours: 40		3	0	0	3

Course Content		Hrs.
Unit 1	Fundamentals of Graph Theory and Graph Analytics: Types of graphs (directed, undirected, weighted), Basic definitions (vertices, edges, neighbours), Applications of graph analytics, Tools and software for graph analysis, Graph representation (adjacency matrix, adjacency list), Graph traversal algorithms (BFS, DFS), Degree, paths, and cycles, Connectivity and components	8
Unit 2	Centrality Measures: Degree centrality, Betweenness centrality, Closeness centrality, PageRank.	4
Unit 3	Graph Neural Networks (GNNS) Basics: Introduction to GNNs, Graph Convolutional Networks (GCNs), Graph Attention Networks, GraphSAGE and Graph Isomorphism Networks (GIN), Message Passing in GNNs, Implementing GNNs with popular frameworks (e.g., PyTorch, TensorFlow)	9
Unit 4	Advanced GNNs and Applications: Graph neural networks for classification, recommendation systems, link prediction and community detection, Ethical considerations in GNN applications	9
Unit 5	Large-Scale Graph Analytics with GNNs: Scalability and performance challenges in GNNs, Distributed GNNs with frameworks like DGL or PyTorch Geometric, Handling large and dynamic graphs with GNNs GNN-based project involving real-world graph data, case studies on GNN applications in various domains (social networks, biology, finance, etc.)	10

References

1.	"Graph Representation Learning" by William L. Hamilton, Rex Ying, and Jure Leskovec
2.	"Graph Convolutional Networks" by Thomas Kipf and Max Welling
3.	"Networks: An Introduction" by Mark Newman
4.	"Graph Algorithms" by Shimon Even and Guy Even

Data Analytics				
Prerequisite: Basic understanding of probability and statistics, linear algebra and calculus. A basic knowledge of programming (preferably Python) is essential.	L	T	P	C
Total hours: 42	3	0	0	3

Course Content		Hrs
Unit 1	Data Understanding and Preparation Introduction, Reading data from various sources, Data visualization, Distributions and summary statistics, Relationships among variables, Extent of Missing Data. Segmentation, Outlier detection, Automated Data Preparation, Combining data files, Aggregate Data, Duplicate Removal, Sampling DATA, Data Caching, Partitioning data, Missing Values Data : Gather, extract, analyse, and manipulate data to draw conclusions or insights. With algorithms and coding with dataset available	10
Unit 2	Introduction to Data Mining: Classification- Naïve Bayes, Clustering- K means , Model development & techniques Data Partitioning, Model selection, Model Development Techniques,	10
Unit 3	Neural networks, Decision trees, Logistic regression, Discriminant analysis, Support vector machine, Bayesian Networks, Linear Regression, Logistic Regression, Association rules..	10
Unit 4	Model Evaluation and Deployment Introduction, Model Validation, Rule Induction Using CHAID, Automating Models for Categorical and Continuous targets, Comparing and Combining Models, Evaluation Charts for Model Comparison, Meta Level Modelling, Deploying Model, Assessing Model Performance, Updating a Model. Visualisation	10

References	
1.	Daniel T. Larose and Chantal D. Larose, Discovering Knowledge in Data: An Introduction to Data Mining, 2nd Edition, Wiley, 2014. ISBN: 978-0-470-90874-7
2.	Recommended Reading: Foster Provost and Tom Fawcett, Data Science for Business: What You Need to Know About Data Mining and Data-Analytic Thinking, O'Reilly, 2013. ISBN: 978-1-449-36132-7

Data Compression					
Prerequisite: Object Oriented Analysis and Design		L	T	P	C
Total hours: 42		3	0	0	3
Course Content					Hrs.
Unit 1	Introduction: Compression techniques, lossless compression, lossy compression, measures of performance, modeling and coding.				8
Unit 2	Mathematical preliminaries - Overview, introduction to information theory, models, physical models, probability models, Markov models.				10
Unit 3	Basic Coding Schemes: Statistical Methods - Shannon-Fano Algorithm, Huffman Algorithm, Adaptive Huffman Coding. Arithmetic Coding (Encoding, Decoding, Adaptive Coding). Dictionary Methods - LZ77, LZ78, LZW Algorithms. Case study of lossless compression standards.				10
Unit 4	Lossless Compression standards: zip, gzip, bzip, unix compress, GIF, JBIG. Image and Video Compression: Discrete Cosine Transform, JPEG. Wavelet Methods - Discrete Wavelet Transform, JPEG 2000				8
Unit 5	Motion Compensation, Temporal and Spatial Prediction. MPEG and H.264. Audio Compression: Digital Audio, WAVE, FLAC, MPEG-1/2 Audio Layers.				6
References:					
1.	Khalid Sayood 2012. Introduction to Data Compression (4th ed.). Elsevier				
2.	David Salomon, Giovanni Motta. 2010. Handbook of Data Compression. Springer, London				

Cloud Computing				
Prerequisite: Operating System, Computer Networks	L	T	P	C
Total hours: 40	3	0	0	3

Course Content		Hrs.
Unit 1	Introduction of Cloud Computing: Taxonomy and related technologies, Essential Characteristics, Service and Deployment Models.	8
Unit 2	Virtualization: Types of Virtualization and Hypervisors, Virtualization at Storage, Compute and Network, Hypervisors – Types, Case studies: KVM, Xen, vSphere / ESXi, Hyper-V, VM Provisioning, VM Migration.	8
Unit 3	Architectures: Standards, Orchestration, Provisioning, Portability, Interoperability, Federated Cloud, Case Studies: OpenStack, vCloud, OpenShift, CloudStack	8
Unit 4	Containerization, Containers, Docker, Docker Hub, Docker Swarm, Kubernetes, Mesos, Magnum; Microservices, DevOps – Version control (Git), Automation (Jenkins), Configuration management (Puppet), Testing (Selenium), Monitoring (Nagios)	8
Unit 5	Security: CIA Triad, Vulnerabilities in Cloud, Threats to Infrastructure, Data and Access Control; Identity Management; Multi Tenancy Issues; Attack taxonomy; Intrusion Detection, VM Specific attacks, VM Introspection, Management; Trusted Cloud Initiative;	8

References

1.	K. Hwang, G. C. Fox, and J. Dongarra, Distributed and Cloud Computing, 1st ed.: Morgan Kaufmann, 2011
2.	R. Buyya, J. Broberg, and A. M. Goscinski, Cloud Computing: Principles and Paradigms: Wiley-Blackwell, 2011
3.	S. Dinkar and G. Manjunath, Moving to the Cloud: Developing Apps in the New World of Cloud Computing Syngress Media, U.S., 2012.
4.	W. Stallings, Foundations of Modern Networking: SDN, NFV, QoE, IoT, and Cloud, 1st ed.: Addison-Wesley Professional, 2015.
5.	T. Erl, Z. Mahmood, and R. Puttini, Cloud Computing: Concepts, Technology & Architecture: Prentice Hall/PearsonPTR, 2014.
6.	R. L. Krutz and R. D. Vines, Cloud Security - A Comprehensive Guide to Secure Cloud Computing, Wiley Publishing, 2010

Quantum Computing					
Prerequisite: None		L	T	P	C
Total hours: 42		3	0	0	3
Course Content					Hrs.
Unit 1	Introduction to quantum computing				8
Unit 2	Relevant Linear algebra for quantum computing, Postulates of quantum mechanics,				10
Unit 3	Classical computing, Quantum circuits, Quantum Fourier Transform				10
Unit 4	Quantum search algorithms, Physical realization of quantum computers.				8
Unit 5	Quantum noise, Quantum operations, quantum information and quantum channel				6
References:					
1.	Pittenger A. O., An Introduction to Quantum Computing Algorithms				
2.	Nielsen M. A., Quantum Computation and Quantum Information, Cambridge University Press.				
3.	Benenti G., Casati G. and Strini G., Principles of Quantum Computation and Information, Vol. I: Basic Concepts, Vol II: Basic Tools and Special Topics, World Scientific.				

Parallel and Distributed Computing					
Prerequisite: Programming in C, Data Structures, Operating Systems, Computer Architecture and Organization		L	T	P	C
Total hours: 40		3	0	0	3
Course Content					Hrs
Unit 1	Parallel Computing, Sequential programs, Parallel Programs, Performance Metrics for Parallel Systems, Effect of Granularity on Performance, Scalability of Parallel Systems, Parallel Programming Platforms, Implicit Parallelism, SIMD & MIMD systems, Clusters, Single-Core and Multi-Core Processors, Physical Organization of Parallel Platforms, Cache Coherence, Posix-Threads, problem-Solving using P-threads.				8
Unit 2	Programming Using the Message-Passing Paradigm - MPI Principles of Message Passing Programming; Building blocks (Sending and Receiving Operations); Communication Library calls; Collective communication and Computation library calls, Programming Shared Address Space Platforms – OpenMP, Directive Parallel Programming; The OpenMP programming Model (Concurrent Tasks, Synchronization Constructs, Data Handling); Open libraries; OpenMP-Environment Variables;				10
Unit 3	Parallel Programs, Matrix Computations, Matrix-Vector Multiplication, Matrix-Matrix Multiplication, Solving system of Linear Equations; Parallel Implementation of Sparse Matrix Computations with Vector; Sorting algorithms, Issues in Sorting on Parallel Computers, Bubble Sort and its Variants, Quicksort; Parallelizing Quicksort; Sequential and Parallel Implementation of all-pairs of Shortest Paths Algorithms; Sequential & Parallel Search Algorithms; Depth-First Search Algorithms; Best-First Search Algorithms				8
Unit 4	Programming on Multi-Core Systems with GPU accelerators, An Overview of Brief History of GPUs; An Overview of GPU Programming; An Overview of GPU Memory Hierarchy Features; An Overview of CUDA enabled NVIDIA GPUs, Introduction to CUDA C, Parallel Programming using OpenACC, CUDA APIs, CUDA Libraries for Numerical and Non-Numerical Computations; The OpenCL – Heterogeneous Programming; OpenCL Libraries, The OpenCL Memory Model, Execution Model; Platform and Devices; An Overview of OpenCL API;				6
Unit 5	An Overview of MapReduce, An Overview of MapReduce Programming, An Overview of Hadoop Architecture /Execution (Master/slave, Namenode/Datanode); Hadoop Distributed File System (HDFS), An Overview of Hadoop Components, Hadoop – Control Flow and Data Flow; An overview of Hive (Distributed Data Warehouse); Hbase (Distributed Column based database, PIG –(Data Flow Language), Introduction to Spark, Spark RDD, Machine Learning Using Spark.				8

References	
1.	Ananth Grama, Anshul Gupta, George Karypis, Vipin Kumar: Introduction to Parallel Computing, Second Edition Pearson Education – 2007
2.	Peter Pacheco, An Introduction to Parallel Programming, Morgan Kaufman Publishers, Elsevier (2011)
3.	Jason Sanders, Edward Kandrot, CUDA By Example – An Introduction to General-Purpose GPU Programming, Addison Wesley (2011)
4.	Rohit Chandra, Leonardo Dagum, Dave Kohr, Dror Maydan, Jeff McDonald, Ramesh Menon, Parallel Programming in OpenMP, Academic Press (2001)
5.	Benedict R Gaster, Lee Howes, David R Kaeli Perhaad Mistry Dana Schaa, (2011), Heterogeneous Computing with OpenCL McGraw-Hill, Inc. Newyork
6.	Michael J. Quinn, Parallel Programming in C with MPI and OpenMP McGraw-Hill International Ed (2003)
7.	Aru C Murthy, Vinod Kumar Vavilapalli, Doug Eadline, Joseph Niemiec, and Jeff Markham, Apache Hadoop YARN Moving beyond MapReduce and Batch Processing with Apache Hadoop 2, Addison Wesley, 2014

Parallel and Distributed Computing Lab				
Prerequisite: C Programming, DSA	L	T	P	C
	0	0	2	1

Course Content

1. Implementation of pthreads, problem-solving using pThreads.
2. Problem-solving using openMP
3. Matrix multiplication using task.
4. Problem-solving using MPI, Sending and Receiving Operations
5. Parallel Programs, Matrix Computations, Matrix-Vector Multiplication, Matrix-Matrix Multiplication using MPI.
6. Parallel Implementation of Sparse Matrix Computations with Vector; Sorting Algorithms, Issues in Sorting on Parallel Computers; Bubble Sort and its Variants using GPU Resources.
7. Quicksort; Parallelizing Quicksort; Sequential and Parallel Implementation of all-pairs of Shortest Paths Algorithms; Sequential & Parallel Search Algorithms.
8. Depth-First Search Algorithms; Best-First Search Algorithms. Control Flow graph generation from a given intermediate code.
9. Implementation of MapReduce programs for large scale data handling.
10. Programming on Multi-Core Systems with GPU accelerators.

References

1.	Aru C Murthy, Vinod Kumar Vavilapalli, Doug Eadline, Joseph Niemiec, and Jeff Markham, Apache Hadoop YARN Moving beyond MapReduce and Batch Processing with Apache Hadoop 2, Addison Wesley, 2014
2.	Benedict R Gaster, Lee Howes, David R Kaeli Perhaad Mistry Dana Schaa, (2011), Heterogeneous Computing with OpenCL McGraw-Hill, Inc. Newyork
3.	Jason Sanders, Edward Kandrot, CUDA By Example – An Introduction to General-Purpose GPU Programming, Addison Wesley (2011) .

Parallelizing Compiler					
Prerequisite: Compiler Design		L	T	P	C
Total hours: 42		3	0	0	3
Course Content					Hrs
Unit 1	Introduction – Compilation for parallel machines and automatic detection of parallelism, structure of a parallelizing compiler.				8
Unit 2	Dependence Theory and Practice - Types of dependences, data and control dependencies, dependence analysis, direction vectors, loop carried and loop independent dependences, tests for data dependence and their applicability, construction of data dependence and control dependence graphs.				18
Unit 3	Parallel Code Generation - Automatic extraction of parallelism, representation of iteration spaces of nested loops, loop-based transformations such as loop distribution, loop coalescing, loop interchange and cycle shrinking transformation.				8
Unit 4	Inter-procedural Analysis and Optimization - aliasing information, summary data flow analysis, inter-procedural constant propagation, inter-procedural data dependence analysis and parallelization of call statements.				8

References

1.	Randy Allen, Ken Kennedy: Optimizing compilers for modern architectures. Morgan Kaufmann.
2.	Steven Muchnick : Advanced Compiler Design & Implementation, Morgan Kaufmann.
3.	Hector, Ullman, Widom : Database System Implementation, Pearson.

System on Chip					
Prerequisite: None		L	T	P	C
Total hours: 42		3	0	0	3
Course Content					Hrs.
Unit 1	Transaction-Level Modeling& Electronic System-Level Languages,				8
Unit 2	Hardware Accelerators, Media Instructions, Co-processors				10
Unit 3	System-Level Design Methodology ,High-Level Synthesis (Cto-RTL),				10
Unit 4	Hardware Synthesis and Architecture Techniques Source-Level Optimizations.				8
Unit 5	Scheduling Resource, Binding and Sharing.				6
References:					
1.	De Micheli, editor Special Issue on Hardware/Software Co-design Proceedings of IEEE, Vol 85, No. 3, March 1997				
2.	D. D. Gajski, F. Vahid, S. Narayan, J. Gong :Specification and Design of Embedded Systems, Prentice Hall, Englewood Cliffs, NJ, 1994				
3.	J. Staunstrup and W. Wolf, editors: Hardware/Software Co-Design: Principles and Practice Kluwer Academic Publishers, 1997				
4.	G. DeMicheli, R. Ernst, and W. Wolf, editors, Readings in Hardware/Software Co-Design, Academic Press, 2002.				

Evolving Architectures				
Prerequisite: Operating Systems, Computer Networks, DBMS, Algorithms	L	T	P	C
Total hours: 32	3	0	0	3

Course Content		Hrs
Unit 1	Special, emerging and advanced topics in different areas of Computer Science and Engineering will be covered under this course. • Understand Taxonomy of new Architectures • Understand the Building Blocks of each architecture. • Install the Open-Source Tools • Study the State of the Art • Listen to an Expert (Academia / Industry) • Discuss Survey / Research Papers (Last 5-7 years) • Case Studies of Tool or Simulator • Build some components for a Simple Model as assignment.	8
Unit 2		8
Unit 3		8
Unit 4		8

References	
1.	Research Papers from Journals and Conferences
2.	Technical and Research Reports from Consortiums / Committees
3.	Red Books, White Papers, Request For Comments (RFCs)
4.	Manuals, Guides, Blogs

Evolving Architectures lab				
Prerequisite: Basic Programming Skills.	L	T	P	C
Total hours: 36	0	0	2	1
Course Content				Hrs
The Lab experiments may change based on the course requirements. **Assignment 1- Virtualization:** Install three servers using Xen / KVM hypervisor. Use two of them to install any application server (Apache etc) and any database server (MySQL etc). Create an application (say a simple login) on the third using the other two servers. Access the application from your host OS and another system **Assignment 2- Cluster Setup:** A computer cluster consists of a set of loosely or tightly connected computers that work together so that, in many respects, they can be viewed as a single system. Unlike grid computers, computer clusters have each node set to perform the same task, controlled and scheduled by software. Some experiments related to MPI will be conducted on the cluster. **Assignment 3- Virtual Machine Introspection Tool:** LibVMI is a virtual machine introspection library. This means that it helps you access the memory of a running virtual machine. LibVMI provides primitives for accessing this memory using physical or virtual addresses and kernel symbols. LibVMI also supports accessing memory from a physical memory snapshot, which is helpful for debugging or forensic analysis. LibVMI is written in C. Install LibVMI tool. Access and analyze memory, hardware events, and virtual CPU registers of running virtual machine installed on top of Xen/KVM Hypervisor. **Assignment 4- CloudSim:** To conduction the following experiments: 1) Resource Scheduling (Min-Min Algorithm). 2) Fault Tolerance (Earliest-Deadline-First (EDF) scheduling Algorithm). 3) Live Virtual Machine Migration. **Assignment 5- Cloud Software Setup** **Assignment 6- Hadoop Multinode Setup** **Assignment 7- Hadoop Ecosystem Setup** **Assignment 8- Android Toolkit Setup** **Assignment 9- Internet of Things: Environmental Setup** **Assignment 10- Software Defined Network Setup**				36
References				
1.	Web Links: https://help.ubuntu.com/community/Xen https://help.ubuntu.com/community/KVM/Installation http://www.ubuntu.com/download/desktop http://www.centos.org/download/ https://help.ubuntu.com/community/MpichCluster https://wiki.ubuntu.com/MpichCluster http://www.cloudbus.org/cloudsim/ http://www8.hp.com/us/en/cloud/helion-eucalyptus-overview.html http://docs.openstack.org/icehouse/install-guide/install/yum/content/ https://hadoop.apache.org/releases.html http://www.tutorialspoint.com/hadoop/hadoop_multi_node_cluster.htm https://hadoopecosystemtable.github.io/ https://hadoop.apache.org/ http://developer.android.com/about/android.html https://developer.android.com/sdk/index.html http://xda-university.com/as-a-developer/getting-started-setting-up-android-development-environment http://www.oracle.com/us/solutions/internetofthings/overview/index.html?ssSourceSiteId=ocomcn http://www.nimbits.com/howto_install.jsp https://thingspeak.com/ https://www.paraimpu.com/ http://www.contiki-os.org/ http://networkstatic.net/how-to-build-an-sdn-lab-without-needing-openflow-hardware/ http://networkstatic.net/openflow-openvswitch-and-kvm-sdn-lab-installation-app/			

Distributed Systems				
Prerequisite: None	L	T	P	C
Total hours: 42	3	0	0	3

Course Content		Hrs.
Unit 1	Introduction to Distributed Systems, OS and Advanced OS, various distributed systems, Trends in Distributed System and challenges, Networking: network protocols, point-to-point communication. Introduction – Clocks, events and process states – Synchronizing physical clocks Logical time and logical clocks – Global states, Limitations, Lamport's logical clock, vector clock, causal ordering, global state, Cuts. Distributed Mutual Exclusion: Lamport, Recart-agrawala, and Maekawa's algorithms; Suzuki-kasami broadcast algorithm, and Raymond's tree based algorithm , Elections algorithms	8
Unit 2	Transactions and Concurrency Control– Transactions -Nested transactions – Locks – Optimistic concurrency control – Timestamp ordering – Atomic Commit Distributed transactions: two phase commit, three-phase commit, ACID/BASE models Techniques of Inter process Communication: the API for internet protocols – External data representation and Multicast communication, Sun RPC: programming and implementation, Network virtualization: Overlay networks. Case study: MPI Remote Method Invocation And Objects: Remote Invocation – Introduction – Request-reply protocols – Remote procedure call – Remote method invocation. Case study: Java RMI – Group communication – Publish-subscribe systems – Message queues – Shared memory approaches – Distributed objects.	10
Unit 3	Case study: Enterprise Java Beans -from objects to components. Distributed Deadlock Detection: Resource Vs. Communication deadlock, Replication, Strategies to handle deadlock, Ho-Ramamoorthy, Path-Pushing, Edge-Chasing, Diffusion Computation-based algorithms. Agreement Protocols: System model, Classification of agreement problems, Solutions to Byzantine Agreement (BA) problems. Distributed Scheduling: Issues in Load Distribution, Components of a load distribution algorithm, Load Distribution Algorithms, V-system, Sprite, and Condor.	10
Unit 4	Network file systems: design, NFS, AFS (scale), DFS & CIFS (cache control), CODA (redundancy) Google File System (GFS), Hadoop Distributed File System (HDFS)Distributed Shared Memory: Algorithms for implementing DSMs, Memory Coherence, and Coherence Protocols, IVY Process Management: Process Migration: Features, Mechanism – Threads: Models, Issues, Implementation.	8
Unit 5	Resource Management: IntroductionFeatures of Scheduling Algorithms –Task Assignment Approach – Load Balancing Approach – Load Sharing Approach Recovery: Classification of failures, Synchronous and Asynchronous Check pointing and Recovery. Fault Tolerance: Commit Protocols, Voting Protocols, Failure Resilient Processes. Protection and Security: Access Matrix Model, Implementation of access matrix, Unix, and Amoeba. Case study-Distributed systems.	6

References:

1.	Andrew S. Tanenbaum, M. V. Steen, "Distributed Systems Principles and Paradigm," 2nd Edition, Pearson
2.	George Coulouris, Jean Dollinmore, Tim Kindberg, Gordon Blair "Distributed SystemsConcepts and Design," 5th Edition, Pearson

3.	M. Singhal & N. Shivaratri, “Advanced Concepts in Operating Systems: Distributed, Database and Multiprocessor Operating Systems”, Tata McGraw Hill, 2015
4.	John Bloomer, “Power Programming with RPC,” O’Reilly & Associates, Inc
5.	Advanced Programming in the Unix Environment by W. Richard Stevens, Addison-Wesley.
6.	Liu M.L., “Distributed Computing, Principles and Applications”, Pearson Education
7.	Distributed Systems - An Algorithmic approach by Sukumar Ghosh.

Deep Learning for NLP					
Prerequisite: A course on Machine Learning or equivalent		L	T	P	C
Total hours: 40		3	0	0	3
Course Content					Hrs
Unit - I	Introduction to NLP and Deep Learning: Multilayer Neural Networks, Back-propagation, Word Vectors: Simple word vector representations: word2vec, GloVe; sentence, paragraph and document representations				10
Unit - II	Sequence modelling: Recurrent Neural Networks and Language Models, Vanishing Gradients problem, GRUs and LSTMs;				8
Unit - III	Machine Translation, Seq2Seq and Attention, Advanced Attention, Transformer Networks and CNNs, Coreference Resolution				8
Unit - IV	Tree Recursive Neural Networks and Constituency Parsing, Advanced Architectures and Memory Networks, Reinforcement Learning for NLP, Semi-supervised Learning for NLP, Future of NLP Models, Multi-task Learning and QA Systems,				10
Unit-IV	Design and Applications of Deep Nets to Language Modelling, Case study of recent state-of-the-arts large language models.				4

References	
1.	Ian Goodfellow and Yoshua Bengio and Aaron Courville. Deep Learning.
2.	Karthiek Reddy Bokka, Shubhangi Hora, Tanuj Jain, Monicah Wambugu, Deep Learning for Natural Language Processing, 2019. O'Reilly pulblication
3.	Recent Literature

Computer Vision					
Prerequisite:		L	T	P	C
Total hours: 40		3	0	0	3
Course Content					Hrs
Unit 1	Introduction to computer vision: Applications of computer vision, basic concepts of image formation, geometric camera models, orthographic and perspective projections, weak perspective projection, intrinsic and extrinsic camera parameters, geometric camera calibration, linear filtering, correlation, convolution				8
Unit 2	Feature detection and matching: Edge detection, interest points and corners, local image features, SIFT, SURF, HoG, LBP, GLCM, etc. Feature matching, bag-of-words, VLAD, RANSAC, Hough transform, image pyramids, 2D transformations				10
Unit 3	Stereo Vision: Stereo camera geometry and epipolar constraints, local methods for stereo matching, global methods for stereo matching, optical flow, structure from motion				8
Unit 4	Machine Learning in Computer Vision: Image recognition, segmentation by clustering, tracking, applications of machine learning in computer vision				6
Unit 5	Deep Learning in Computer Vision: Recognition, detection, segmentation, and activity recognition, introduction to CNNs, evolution of CNN architectures, visualization and understanding CNN Deep Generative Models in Vision: GANs, VAEs, etc. Modern Approaches: Attention models in vision, vision transformer (ViT)				8

References

1.	Forsyth, D. A. and Ponce, J., "Computer Vision: A Modern Hall, 2 nd Ed., 2011
2.	Szeliki, R., "Computer Vision: Algorithms and Applications", Springer, 2011
3.	Hartley, R. and Zisserman, A., "Multiple View Geometry in Computer Vision", Cambridge University Press, 2004
4.	Goodfellow, I., Bengio, Y., and Courville, A., “Deep Learning”, MIT Press, 2016

Computer Vision Lab				
Prerequisite: Fundamental knowledge on image processing, machine learning, and programming skills	L	T	P	C
	0	0	2	1

Course Content

1. Familiarization with various computer vision tools
2. Basic operations on images and videos
3. Linear filtering and convolution
4. Implementation of different image transforms
5. Implementation of various feature descriptors (SIFT, SURF, HoG, LBP, GLCM, etc.)
6. Edge detection, line detection and corner detection
7. Implementation of feature matching algorithms
8. Segmentation by clustering
9. Implementation of neural network architectures
10. Implementation of CNN architectures for various tasks such as classification, segmentation, object detection, etc., and transfer learning
11. Implementation of GAN and ViT models
12. Mini project

References:

1.	Richard Szeliski, Computer Vision: Algorithms and Applications, Springer, 2010
2.	Bishop, Christopher M, Pattern Recognition and Machine Learning, Springer, 2006
3.	Ian Goodfellow, Yoshua Bengio, Aaron Courville, Deep Learning, 2016

Biometrics					
Prerequisite:A basic knowledge of statistics, linear algebra, and programming is expected.		L	T	P	C
Total hours: 42		3	0	0	3
Course Content					Hrs
Unit 1	Introduction: Person recognition, Biometric systems, Biometric functionalities, biometrics system errors, the design cycle of biometric systems				6
Unit 2	Fingerprint recognition: friction ridge patterns, finger print acquisition, feature extraction and matching, palm prints				8
Unit 3	Face recognition: image acquisitions, face detection, feature extraction and matching, handling pose, illumination and expression variations				8
Unit 4	Iris recognition: image acquisition, Iris segmentation, Iris normalization, Iris encoding and matching, Iris quality assessment techniques				6
Unit 5	Additional Biometric Traits: Ear, Gait, Hand geometry, Soft biometrics Multimiometrics: sources of multiple evidence, fusion levels: sensor, feature, score, rank and decision level fusion				8
Unit 6	Security of biometric systems: adversary attacks, attacks at user interface, attacks on biometric processing, attacks on template database				6

References

1.	Introduction to Biometrics, Anil K Jain Arun Ross, Springer
2.	The Science of Biometrics, Ravindra Das, Springer
3.	Practical Biometrics, Julian Ashbourn, Springer
4.	Introduction to Biometrics, Anil K Jain Arun Ross, Springer

Biometrics Lab				
Prerequisite: A basic knowledge of statistics, linear algebra, and programming is expected.	L	T	P	C
	0	0	2	1
Course Content				
Familiarization with image processing toolbox, implementation of fingerprint recognition algorithms and systems, feature extraction and matching algorithms, design of face recognition systems, face detection, implementation of iris recognition systems, design of multimodal biometric system using fingerprint, face, speech, etc., fusion strategies, design of biometric system using other biometric traits (ear, gait, Hand geometry, etc.), Security of biometric systems, Mini project				

References

1.	Anil K. Jain, Arun Ross, Introduction to Biometrics, Springer
2.	Ravindra Das, The Science of Biometrics, Springer
3.	Julian Ashbourn, Practical Biometrics, Springer

Large Language Models					
Prerequisite: NLP, Basics of Machine Learning		L	T	P	C
Total hours: 40		3	0	0	3
Course Content					Hrs
Unit 1	Introduction to Large Language Models, What are Large Language Models, History and Development, Neural Networks and Transformers architecture				6
Unit 2	Model Architecture, BERT (encoder-only models), T5 (encoder-decoder models), GPT-3 (decoder-only models), Prompting for few-shot learning, Prompting as parameter-efficient fine-tuning, In-context learning, Calibration of prompting LLMs, Reasoning (Chain of thought Prompting elicits reasoning in large language models), Hyperparameters and Optimization.				7
Unit 3	Language Models as Knowledge Bases?, How Much Knowledge Can You Pack Into the Parameters of a Language Model?, Documenting Large Web Text Corpora: A Case Study on the Colossal Clean Crawled Corpus, Training Compute-Optimal Large Language Models, Extracting Training Data from Large Language Models, RealToxicityPrompts: Evaluating Neural Toxic Degeneration in Language Models.				7
Unit 4	Self-Diagnosis and Self-Debiasing: A Proposal for Reducing Corpus-Based Bias in NLP, Switch Transformers: Scaling to Trillion Parameter Models with Simple and Efficient Sparsity, Improving language models by retrieving from trillions of tokens, Training language models to follow instructions with human feedback,				8
Unit 5	Evaluating Large Language Models Trained on Code, Flamingo: a Visual Language Model for Few-Shot Learning, Alexander Rush (Cornell/Hugging Face) Multitask Prompted Training for Zero-Shot Models, AI Alignment + open discussion				6
Unit 6	Ethical Considerations, Bias and Fairness, Privacy Concerns, Misuse and Mitigation, Case studies and discussions, Applications of Large Language Models, Natural Language Understanding and Generation, Chatbots and Virtual Assistants, Sentiment Analysis, Hands-on: Building a language model application				6
References					
1.	GPT-3: Building Innovative NLP Products Using Large Language Models by Sandra Kublik (Author), Shubham Saboo (Contributor).				
2.	Transformers for Natural Language Processing: Build, train, and fine-tune deep neural network architectures for NLP with Python, Hugging Face, and OpenAI's GPT-3, ChatGPT, and GPT-4 by Denis Rothman				
3.	Natural Language Processing with Python by Edward Loper, Ewan Klein, and Steven Bird				
4.	"Deep Learning" by Ian Goodfellow and Yoshua Bengio				

Cloud Security					
Prerequisite: Computer Networks, Operating System		L	T	P	C
Total hours: 30		3	0	0	3
Course Content					Hrs
Unit 1	Introduction of Cloud Computing: Taxonomy and related technologies, Essential Characteristics, Service and Deployment Models. Virtualization: Types of Virtualization and Hypervisors, Virtualization at Storage, Compute and Network, Hypervisors (Types and Case studies), Virtual Machine Provisioning, Virtual Machine Migration.				10
Unit 2	Architectures: Standards, Orchestration, Provisioning, Portability, Interoperability, Federated Cloud, Security: CIA Triad, Vulnerabilities in Cloud, Threats to Infrastructure, Data and Access Control; Identity Management; Multi Tenancy Issues; Attack taxonomy; Intrusion Detection, VM Specific attacks, VM Introspection, Management; Trusted Cloud Initiative of Cloud Security Alliance (CSA).				10
Unit 3	Forensics: NIST Forensics Reference Architecture, Forensic Science Challenges, Architectural Issues, Evidence Collection and Analysis, Anti-Forensics, Incident Response, Standards and Framework				10

References

1.	K. Hwang, G. C. Fox, and J. Dongarra, Distributed and Cloud Computing, 1st ed.: Morgan Kaufmann, 2011
2.	R. Buyya, J. Broberg, and A. M. Goscinski, Cloud Computing: Principles and Paradigms: Wiley-Blackwell, 2011
3.	S. Dinkar and G. Manjunath, Moving to the Cloud: Developing Apps in the New World of Cloud Computing Syngress Media, U.S., 2012.
4.	W. Stallings, Foundations of Modern Networking: SDN, NFV, QoE, IoT, and Cloud, 1st ed.: Addison-Wesley Professional, 2015.
5.	P. Mishra, E.S. Pilli, R.C. Joshi, "Cloud Security: Attacks, Techniques, Tools, and Challenges", 1sr Ed., Chapman and Hall/CRC.

Digital Forensics					
Prerequisite: Operating Systems, Computer Networks & Security		L	T	P	C
Total hours: 42		3	0	0	3
Course Content					Hrs.
Unit 1	File System Forensics: Duplicating hard disks for "dead analysis", reading hidden data on a disk's Host Protected Area (HPA), Direct versus BIOS access, dead versus live acquisition				8
Unit 2	Disk partitions - DOS, Apple, and GPT partitions, BSD disk labels, Sun Volume; multiple disk volumes - RAID and disk spanning.				10
Unit 3	Analyzing FAT, NTFS, Ext2, Ext3, UFS1, and UFS2 file systems, Finding evidence: File metadata, recovery of deleted files, Using The Sleuth Kit (TSK), Autopsy Forensic Browser, and related open source tools				10
Unit 4	Web Forensics: network-based evidence in Windows and Unix environments, Reconstructing Web browsing, email activity, Tracing domain name ownership and the source of e-mails				8
Unit 5	System Forensics: Windows Registry changes, Duplicating and analyzing the contents of PDAs and flash memory devices Electronic document, computer image verification and authentication.				6
References:					
1.	Brian Carrier. File System Forensic Analysis, Addison Wesley				
2.	Chris Prosise, Kevin Mandia. Incident Response and Computer Forensics, McGraw Hill. Course Technology.				
3.	Linda Volonino, Reynaldo Anzaldua, and Jana Godwin. Computer Forensics: Principles and Practices, Prentice Hall.				
4.	Keith J. Jones, Richard Bejtlich, and Curtis W. Rose. Real Digital Forensics: Computer Security and Incident Response, Addison Wesley.				
5.	Vacca, John R., Computer Forensics Computer Crime Scene Investigation, Charles River Media.				
6.	Nelson, Phillips, Enfinger, Steuart. Guide to computer Forensics and Investigation				

Embedded System Security					
Prerequisite: None		L	T	P	C
Total hours: 42		3	0	0	3
Course Content					Hrs.
Unit 1	Security Flaws and Attacks in Embedded systems: Code injection, Invasive and Non invasive physical and logical attacks				8
Unit 2	Defenses Against Code Injection Attacks: Methods using Address Obfuscation and Software Encryption, Anomaly Detection.				10
Unit 3	Safe Languages, Code Analyzers Compiler, Library, and Operating System Support for embedded systems				10
Unit 4	Security, Control Flow Checking, IP Protection: Encryption of IP Cores, additive and Constraint-Based watermarking.				8
Unit 5	Implementation of DES 3DES, AES, RC4, MD5, RSA algorithms				6
References:					
1.	Security in Embedded Hardware				

Intrusion Detection				
Prerequisite: None	L	T	P	C
Total hours: 42	3	0	0	3

Course Content		Hrs.
Unit 1	Introduction- Intrusion Detection System (IDS), Intrusion Prevention System (IPS).	8
Unit 2	Unauthorized access – buffer overflow, packet fragmentation, out-of-spec packets Review of Network protocol – TCP/IP, Intrusion detection through tcpdump	10
Unit 3	IDS and IPS – Architecture and internals. Malicious and non-malicious traffic, IP headers, TCP, UDP and ICMP protocols and header formats.	10
Unit 4	Header information to detect intrusion, logs and their analysis.	6
Unit 5	IDS through reaction and response Intrusion analysis – data correlation, tools, SNORT- A case study	8

References:

1.	Matt Fearnow, Stephen Northcutt, Karen Frederick, and Mark Cooper. Intrusion Signatures and Analysis, SAMS.
2.	Carl Endorf, Gene Schultz, Jim Mellander, Intrusion Detection and Prevention, McGraw Hill
3.	Paul E. Proctor. The Practical Intrusion Detection Handbook, Prentice Hall.
4.	Stephen Northcutt and Judy Novak. Network Intrusion Detection, SAMS.

Cryptography					
Prerequisite:		L	T	P	C
Total hours: 40		3	0	0	3
Course Content					Hrs.
Unit 1	Basic Concepts: Information theoretic vs. computational security. One way functions, Pseudo randomness generators and functions, Permutations, hash functions.				8
Unit 2	Private-key encryption using pseudo randomness. Private-key authentication. – Public key encryption (and number theory). Public key authentication.				12
Unit 3	Interactive protocols: Touch of complexity theory, Interactive proof systems; 0-knowledge proof systems,0-knowledge authentication, Electronic cash; non-interactive zero-knowledge.				12
Unit 4	Oblivious transfer: Definitions, constructions, and applications, Secure Multiparty computations, Database (differential) privacy. – Proofs of work – Block-chain consensus protocols.				8
References:					
1.	Introduction to Modern Cryptography: Principles and Protocols, by Jonathan Katz and Yehuda Lindell				
2.	A Graduate Course in Applied Cryptography by Dan Boneh and Victor Shoup				
3.	The Joy of Cryptography by Mike Rosulek.				
4.	OdedGoldreich: Foundations of Cryptography Vol 1 and Vol 2				

Blockchain Technologies					
Prerequisite: Nil		L	T	P	C
Total hours: 35		3	0	0	3
Course Content					Hrs
Unit 1	Introduction to blockchain- Distributed Ledger Technology, Decentralization, Problems in Traditional Money transfer system, Digital Crypto currency, Bitcoin nuts and bolts, Generic elements of Blockchain, Bitcoin Network and Architecture, Block and transactions in a Blockchain, Advantages over Traditional Databases, Mining Mechanism, Types of Blockchain: Public, Private, Consortium, Hybrid				6
Unit 2	Cryptography: Elliptic Curve Cryptography, Hash Functions, Merkle Tree, Merkle Patricia Trie, Digital Signature, Wallets and Keys, User Addresses and Privacy CRYPTO CURRENCY History, Distributed ledger, Creation of Coins, Double spending,				3
Unit 3	Mechanics of Bitcoin, Bitcoin protocols, Transaction in Bitcoin Network, AltCoins, Ethereum, Transactions in Ethereum, EVM, Accounts, Transactions, Gas, Fees, Smart Contracts, Wallets managing and protecting crypto assets, Types of Wallets, different ways of storing Bitcoin keys, security measures, Tokenizing, Risk and challenges,				8
Unit 4	Bitcoin Mining and consensus –definition, working of Consensus Mechanism, Byzantine Generals Problem, Nakamoto consensus, Properties of consensus mechanism , incentives in consensus,Types of Consensus Algorithms, Proof of Work (PoW), Proof of Stake (PoS), Delegated Proof of Stake (DPoS), Proof of Importance (PoI), Proof of Capacity (PoC) ,The Proof of Elapsed Time (PoET), Hybrid Proof of Activity (PoA), Proof of Authority (PoA), Proof of Burn (PoB) Byzantine Fault Tolerance (BFT), and other flavours of consensus mechanisms , Pros and Cons of Consensus Mechanisms, sybil resistance, Security analysis of various Consensus Mechanisms				9
Unit 5	Ethereum Syntax &, Structure, Decentralized Apps (dApp), EVM, and the Ethereum blockchain, Eth 2.0, Sharding Chains ,Smart Contract, , MetaMask, Blockchain-based IoT Applications, Hyperledger, Components of Ethereum Ecosystem Smart contract on ethereum, Setting up Ethereum Node using Geth Client, Smart Contracts and DApps, Programming in solidity Truffle, Ganache CLI, Metamask, Remix, Solidity, Writing and Deploying Smart Contracts in Solidity, Connection to Web3.js Library, Vulnerabilities in Smart Contracts, Attacks, Prevention of Attacks, Decentralized Autonomous Organization (DAO), Building an Initial Coin Offering (ICO). Privacy and Scaling the blockchain, blockchain interoperability, future of blockchains				6
Unit 6	Use Cases and applications in Cryptocurrency and Other Sectors like Finance, Voting System, and Healthcare, Networks , Bitcoin: A Peer-to-Peer Electronic Cash System, Supply Chain Management (SCM) etc				3

References

1.	Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller, Steven Goldfede, “Bitcoin and Cryptocurrency Technologies”, Princeton University Press, 2016.
2.	Lantz, Lorne, and Daniel Cawrey, “Mastering Blockchain: Unlocking the Power of Cryptocurrencies, Smart Contracts, and Decentralized Applications” O’Reilly Media,
3.	Imran Bashir, Mastering Blockchain: Distributed Ledger Technology, decentralization, and smart contracts explained, Packt Publishing Ltd, March 2018

Advanced Algorithms					
Total Hours		L	T	P	C
42		3	0	0	3
Prerequisite: *Data Structures, Design and Analysis of Algorithms, C programming*					
Course Content					Hrs
Unit 1	***Review of Data Structures*** **-** Height balanced trees, AVL, Red-black trees, splay trees, Binomial and Fibonacci heaps, treaps, suffix tree, Range minimum query, Aho-Corasick automata, Hash tables, Tries, van Emde boes tree.				8
Unit 2	***String Algorithms:*** Exact String Matching: Rabin-Karp, KMP, Boyer Moore; Inexact string matching: Edit distance, Levenshtein distance computing algorithm. ***Computational Geometry:*** Convex Hull. Line-segment Intersection. Sweep Lines. Voronoi Diagrams. Range Trees. Optimal polygon triangulation.				8
Unit 3	***Linear Programming:*** Formulation of Problems as Linear Programs. Duality. Simplex, Interior Point, and Ellipsoid Algorithms. ***Online Algorithms:*** Ski Rental. River Search Problem. Paging. The k-Server Problem. List Ordering and Move-to-Front. ***Parallel Algorithms:*** PRAM. Pointer Jumping and Parallel Prefix. Bitonic sorting, Odd-even sorting, Maximal Independent Set.				10
Unit 4	***Approximation Algorithms:*** Greedy Approximation Algorithms. Dynamic Programming and Weakly Polynomial-Time Algorithms. Linear Programming Relaxations. Randomized Rounding. Vertex Cover, Wiring, and TSP. Fixed-Parameter Algorithms - Parameterized Complexity. Kernelization. Vertex Cover. ***Probabilistic algorithms:*** Primality testing, Integer factorization, ***Randomized algorithms:*** Monte Carlo – mincut, Las Vegas – quicksort				8
Unit 5	***Complexity classes*** **-** NP-Hard and NP-complete Problems - Cook's theorem NP completeness reductions – SAT, 3SAT, vertex cover, Independent Set, Hamiltonian cycle, travelling salesman.				8
References					
1.	Cormen, Leiserson, Rivest: Introduction to Algorithms, Prentice Hall Of India.				
2.	Aho A.V , J.D Ulman: *Design And Analysis Of Algorithms*, Addison Wesley				
3.	Jon Kleinberg And Éva Tardos: Algorithm Design, Pearson.				
4.	Motwani And Raghavan: *Randomized Algorithms*, Cambridge University Press				
5.	Vaizirani: *Approximation Algorithms*, Springer Verlag				
6.	Papadimitriou, Steiglitz: *Combinatorial Optimization: Algorithms And Complexity*, Phi.				

Advanced Algorithms Lab				
Prerequisite: Data Structures, Design and Analysis of Algorithms, C programming	L	T	P	C
Total hours: 28	0	0	0	2

Course Content	Hrs.
The following proposed coverage are broad guiding areas lab. The programs mentioned here just sample programs and they are just for reference purpose. The instructor offering the course in consultation with the theory offered can adopt further variations in tune with concerned theory course. 1. AVL Tree Operations (1 lab): Implement and demonstrate insertion and deletion operations on an AVL tree. 2. Red-Black Tree Operations (1 lab): Create a red-black tree and showcase insertion and deletion operations. 3. Splay Tree Operations (1 lab): Implement a splay tree and demonstrate splaying after different operations. 4. Binomial Heap Operations (1 lab): Develop a binomial heap and perform insert, delete, and merge operations. 5. Fibonacci Heap Operations (1 lab): Create a Fibonacci heap and demonstrate insert, extract-min, and decrease-key operations. 6. Suffix Tree Construction (1 lab): Construct a suffix tree for a given input string. 7. Range Minimum Query (RMQ) (1 lab): Implement a data structure to perform efficient RMQ operations. 8. Aho-Corasick Automata (1 lab): Implement the Aho-Corasick string matching algorithm for multiple string patterns. 9. Hash Table Operations (1 lab): Create a basic hash table and perform insert, search, and delete operations. 10. Trie Operations (1 lab): Implement a trie data structure and perform insertion, search, and deletion for strings. 11. van Emde Boas Tree Operations (1 lab): Create a van Emde Boas tree and demonstrate insert, delete, and search operations. 12. Dynamic Programming - Edit Distance (1 lab): Implement the dynamic programming algorithm for computing edit distance between two strings. 13. Convex Hull - Graham's Scan (1 lab): Implement the Graham's Scan algorithm to find the convex hull of a set of points. 14. Line-Segment Intersection (1 lab): Implement a program to detect line-segment intersections in a set of line segments.	

References:	
1.	"Introduction to Algorithms" by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein
2.	"Algorithms" by Robert Sedgewick and Kevin Wayne
3.	"Advanced Data Structures" by Peter Brass
4.	"Approximation Algorithms" by Vijay V. Vazirani

Advances in Compiler Design					
Prerequisite: Basic course in Compiler Design		L	T	P	C
Total hours: 42		3	0	0	3
Course Content					Hrs.
Unit 1	Modern Compiler Design – Structure of Compilers for Modern Programming Languages, Cross Compiler, Just-In-Time (JIT) and Adaptive Compilation				8
Unit 2	Runtime System Architectures. Parser Development - LR Parsers and LR Grammars – Design and Implementation.				10
Unit 3	Parser and Ambiguity, Conflict Resolution, Lex and Yacc Tools. Optimizing Compiler - Control-flow Analysis, Control-flow Graphs, Basic Blocks.				10
Unit 4	Data-flow Analysis Methods, Dependence Analysis, Global Optimizations, Loop Optimizations.				8
Unit 5	Peephole Optimization and Optimal Code Generation, Data Dependence Analysis in Loops, Loop Scheduling.				6
References:					
1.	Aho, Lam, Sethi and Ullman: Compilers – Principles, Techniques and Tools, Pearson Education 2. 3. 4.				
2.	Steven Muchnick : Advanced Compiler Design & Implementation, Morgan Kaufmann				
3.	Holub: Compiler Design in C, Prentice Hall India.				
4.	Keith Cooper and Linda Torczon : Engineering a Compiler, Morgan Kaufmann.				

Advances in Compiler Design Lab

Prerequisite: : Compiler Design	L	T	P	C
	0	0	2	

Course Content

1. Programming exercises on construction of phases of a typical compiler.
2. Programming exercises on LR Parsers construction.
3. Programming exercises using Lex and Yacc tools.
4. Programming exercises on Control-flow Analysis and Control-flow Graphs.
5. Programming exercises on Data-flow Analysis Methods.
6. Programming exercises on Loop Optimizations and Data Dependence Analysis in Loops.

References

1.	Aho, Lam, Sethi and Ullman: Compilers – Principles, Techniques and Tools, Pearson Education.
2.	Steven Muchnick : Advanced Compiler Design & Implementation, Morgan Kaufmann.
3.	Holub: Compiler Design in C, Prentice Hall India.
4.	Keith Cooper and Linda Torczon : Engineering a Compiler, Morgan Kaufmann.

Advanced Database Systems					
Prerequisite: : Database Information Systems		L	T	P	C
Total hours: 42		3	0	0	3
Course Content					Hrs
Unit 1	Query Processing and Optimization – Implementation of Database operations, External Sorting, Size Estimations, Equivalence Rules, Heuristic-based Optimization, Materialized Views, Incremental View Maintenance.				14
Unit 2	Transaction Processing and Implementation - Concurrency Control Protocols, Two-phase Lock Protocol and its variants, Deadlock Prevention and Detection schemes and implementation, Timestamp-based Ordering Protocol,Log-based Recovery Management.				12
Unit 3	Modern Database Systems - Database System Architectures, Distributed Database Systems, Parallel Databases, Times in Databases, Multimedia Databases.				8
Unit 4	Distributed Databases - Data Storage, Global Catalog, Distributed Transaction Processing, Two-Phase Commit Protocol, Distributed Query Processing.				8

References	
1.	Silberschatz, Korth, Sudarshan: Database System Concepts, McGrall Hill.
2.	Elmasri and Navathe: Fundamentals of Database Systems, 3rd Edition, Addison Wesley.
3.	Hector, Ullman, Widom: Database System Implementation, Pearson.
4.	Ceri and Pelagatti: Distributed Databases – Principles and Systems, McGraw Hill.

Advanced Database Systems Lab				
Prerequisite: : Database Information Systems	L	T	P	C
	0	0	2	
Course Content				
1. Programming exercises on Query Processing and Implementation of Database operations. 2. Programming exercises on Query Optimization – Cost-based and Heuristic-based Optimization. 3. Programming exercises on Transaction Processing. 4. Programming exercises on Concurrency Control Protocols. 5. Programming exercises on Log-based Recovery Management. 6. Programming exercises on Distributed Transaction Processing, and Distributed Query Processing.				

References

1.	Silberschatz ,Korth, Sudarshan : Database System Concepts, McGrall Hill.
2.	Elmasri and Navathe : Fundamentals of Database Systems, 3rd Edition, Addison Wesley.
3.	Hector, Ullman, Widom : Database System Implementation, Pearson.
4.	Ceri and Pelagatti : Distributed Databases – Principles and Systems, McGraw Hill.

Software Testing and Validation					
Prerequisite: Software Engineering		L	T	P	C
Total hours: 42		3	0	0	3
Course Content					Hrs
Unit 1	**Introduction**: Software Testing, Importance of testing, Roles and Responsibilities, Testing Principles, Attributes of Good Test, V-Model, Test Case Generation, SDLC Vs STLC.				8
Unit 2	**Types of Testing:** Unit Testing, Integration Testing, System Testing, Regression Testing, Acceptance Testing, Functional/Non Functional Testing, Static and Dynamic Testing **Categorization of testing methods**: Manual Testing, Automation Testing and Automated Testing Vs. Manual Testing, Testing Tools.				6
Unit 3	**Non Functional Testing:** Performance Testing, Load Testing, Security Testing, Scalability Testing, Compatibility Testing, Stress Testing, Installation Testing.				6
Unit 4	**Software Testing Methodologies:** Validation & Verification, White Box Testing, Black Box Testing, Grey Box Testing. White/Glass Box Testing: Statement Coverage Testing, Branch Coverage Testing, Path Coverage Testing, Conditional Coverage Testing, Loop Coverage Testing, Mutation testing, Data Flow Testing. Black Box Testing: Boundary Value Analysis, Equivalence Class Partition, State Based Testing, Cause Effective Graph, Decision Table.				6
Unit 5	**Software Testing Life Cycle:** Requirements Analysis, Test Planning, Objective, Scope of Testing, Schedule, Approach, Roles & Responsibilities, Assumptions, Risks & Mitigations, Entry & Exit Criteria, Test Automation, Deliverables.				8
Unit 6	**Test Cases Design**: Write Test cases, Review Test cases, Test Cases Template, Types of Test Cases, Difference between Test Scenarios and Test Cases, Test Oracle, Test Environment setup, Understand the SRS, Hardware and software requirements, Test Data.				8
References					
1.	A.P. Mathur, " Foundations of Software Testing", Pearson publications.				
2.	Naresh Chauhan , “Software Testing Principles and Practices ” Oxford University Press, New Delhi.				
3.	Srinivasan Desikan and Gopalaswamy Ramesh, “Software Testing – Principles and Practices”, Pearson Education.				

Real Time Systems				
Prerequisite: None	L	T	P	C
Total hours: 40	3	0	0	3

	Course Content	Hrs.
Unit 1	Introduction: Definition, Typical Real Time Applications; Digital Control, High Level Controls, Signal Processing etc., Release Times, Deadlines, and Timing Constraints, Hard Real Time Systems and Soft Real Time Systems, Reference Models for Real Time Systems: Processors and Resources, Temporal Parameters of Real Time Workload, Periodic Task Model, Precedence Constraints and Data Dependency	7
Unit 2	Real Time Scheduling: Common Approaches to Real Time Scheduling: Clock Driven Approach, Weighted Round Robin Approach, Priority Driven Approach, Dynamic Versus Static Systems, Optimality of Effective-Deadline-First (EDF) and Least-Slack-Time-First (LST) Algorithms, Offline Versus Online Scheduling, Scheduling Aperiodic and Sporadic jobs in Priority Driven and Clock Driven Systems	8
Unit 3	Resources Access Control: Effect of Resource Contention and Resource Access Control (RAC), Nonpreemptive Critical Sections, Basic Priority-Inheritance and Priority-Ceiling Protocols, Stack Based Priority-Ceiling Protocol, Use of Priority-Ceiling Protocol in Dynamic Priority Systems, Preemption Ceiling Protocol, Access Control in Multiple-Unit Resources, Controlling Concurrent Accesses to Data Objects	8
Unit 4	Multiprocessor System Environment: Multiprocessor and Distributed System Model, Multiprocessor Priority-Ceiling Protocol, Schedulability of Fixed Priority End-to-End Periodic Tasks, Scheduling Algorithms for End-to-End Periodic Tasks, End to-End Tasks in Heterogeneous Systems, Predictability and Validation of Dynamic Multiprocessor Systems, Scheduling of Tasks with Temporal Distance Constraints.	9
Unit 5	Real Time Communication: Model of Real Time Communication, Soft and Hard RTCommunication systems, Priority-Based Service and Weighted Round-Robin Service Disciplines for Switched Networks, Medium Access Control Protocols forBroadcast Networks, Internet and Resource Reservation Protocols, Real Time Protocols, Communication in Multicomputer System. An Overview of Real Time Operating Systems and Databases: Features of RTOS, UNIX as RTOS, POSIX Issues, Temporal Consistency, Concurrency Control.	8

References:	
1.	Real Time Systems: Theory and Practice – Mall Rajib, Pearson Education, 2009
2.	Real-Time Systems: Scheduling, Analysis, and Verification – Albert M. K. Cheng, Wiley, 2002.
3.	H. Kopetz, "Real time systems: Design Principles for distributed embedded applications”, Springer Publications, 2011.
4.	Douglass, Real Time UML: Advances in the UML for Real-Time Systems, 3/e, AddisonWesley, 2004.
5.	Awad, Kuusela& Ziegler, Object-Oriented Technology for Real Time Systems: A Practical Approach Using OMT and Fusion, l/e, Pearson Education, 1996.
6.	Ward & Mellor, Structured Development for Real-Time Systems, Vol. III: Implementation Modeling Techniques, Prentice Hall, 1986.

Wireless Sensor Networks					
Prerequisite: None		L	T	P	C
Total hours: 42		3	0	0	3
Course Content					Hrs.
Unit 1	Introduction: Introduction to adhoc/sensor networks: Key definitions of adhoc/ sensor networks, unique constraints and challenges, advantages of adhoc/sensor network, driving applications, issues in adhoc wireless networks/sensor network, data dissemination and gathering, Historical Survey of Sensor Networks				8
Unit 2	Basic Architectural Framework: Traditional layered stack, Cross-layer designs, Sensor network architecture, Physical Layer, Basic Components, Hardware Platforms: Motes, Sensor Devices, Types of Sensors, Sensor's Specification				8
Unit 3	MAC Protocols: Fundamentals of MAC protocols - Low duty cycle protocols and wakeup concepts - Contention Based protocols - Schedule-based protocols - SMAC - BMAC - Traffic-adaptive medium access protocol (TRAMA) - The IEEE 802.15.4 MAC protocol. Routing Protocols: Issues in designing a routing protocol, classification of routing protocols, table-driven, on-demand, hybrid, flooding, hierarchical, and power aware routing protocols.				12
Unit 4	Sensor network security: Security Requirements, Issues and Challenges in Security Provisioning, Network Security Attacks, Layer wise attacks in wireless sensor networks, possible solutions for jamming, tampering, black hole attack, flooding attack. Key Distribution and Management.				8
Unit 5	Secure Routing – SPINS, reliability requirements in sensor networks. Programming in WSNs: Challenges and limitations of programming WSNs, Introduction to TinyOS, - Programming in Tiny OS using NesC, Emulator TOSSIM, Open research issues				6
References:					
1.	Feng Zhao, Leonidas Guibas, “Wireless Sensor Network”, Elsevier, 1st Ed. 2004 (ISBN: 13-978-1-55860-914-3)				
2.	Kazem, Sohraby, Daniel Minoli, TaiebZnati, “Wireless Sensor Network: Technology, Protocols and Application”, John Wiley and Sons 1st Ed., 2007 (ISBN: 978-0-471-74300-2).				
3.	Raghavendra, Cauligi S, Sivalingam, Krishna M., ZantiTaieb, “Wireless Sensor Network”, Springer 1st Ed. 2004 (ISBN: 978-4020-7883-5).				
4.	E. H. Callaway, Jr. E. H. Callaway, Wireless Sensor Networks Architecture and Protocols, CRC Press, 2009				

Internet of Things					
Prerequisite: Nil		L	T	P	C
Total hours: 42		3	0	0	3
Course Content					Hrs
Unit 1	Introduction: Internet of Things and Connected Products, IoT paradigm, Smart objects, Goal orientation, Convergence of technologies; Business Aspects of the Internet of Things. IoT Architectures and Protocols: Importance, Communication models in IoT, Layers in IoT architecture, Role of protocols in IoT communication.				6
Unit 2	Wireless Technologies for IoT: Wi-Fi and 802.15.x family - features, range, power consumption; Zigbee: Network topology, mesh networking, Zigbee stack, LoRaWAN, SigFox, Cellular technologies for IoT (2G, 3G, 4G, NB-IoT, 5G). Latest developments in communication technologies.				8
Unit 3	IoT Network Topologies: Overview (star, mesh, hybrid, etc.), selection based on advantages and limitations. Network Protocols: overview of IoT network protocols. IPv6 and its significance in IoT addressing, 6LoWPAN- IPv6 over Low-power Wireless Personal Area Networks, Header Compression. RPL-overview and operation, Case studies of network protocols in IoT deployments.				10
Unit 4	IoT Application Protocols: Introduction (MQTT, CoAP, HTTP, etc.), Comparison of IoT protocols (features, performance, scalability), MQTT - Concepts, message structure, QoS levels, CoAP - Principles, RESTful architecture, resource discovery, HTTP in IoT- Web services, REST APIs, JSON/XML data formats. Data: OMA Lightweight M2M (LwM2M) protocol, OPC Unified Architecture (OPC UA), BACnet, Modbus. Data Protocols and Formats: IoT data formats (JSON, XML, CBOR, Protocol Buffers). IoT Standards and Interoperability				10
Unit 5	IoT Security Protocols: Security challenges and threats in IoT, Secure communication protocols for IoT (DTLS, TLS, IPsec), Authentication and access control in IoT, Security protocols for device management and firmware updates, Privacy protection and data encryption in IoT.				4
Unit 6	Emerging Trends: Blockchain, 5G and its impact, IoT and edge-cloud integration, latest advancements in IoT architectures and protocols.				4

References

1.	The Internet of Things: Key Applications and Protocols, David Boswarthick, OlivierHersent, and Omar Elloumi, Wiley.
2.	Building the Internet of Things with IPv6 and MIPv6, Daniel Minoli, Wiley.
3.	Architecting the Internet of Things, Dieter Uckelmann, Mark Harrison, and Florian Michahelles, Springer.
4.	Latest research articles.

Software Engineering					
Prerequisite: :Nil		L	T	P	C
Total hours: 42		3	0	0	3
Course Content					Hrs
Unit 1	**Introduction to Software Engineering**: The evolving Role of Software Engineering, The Changing Nature of Software, Legacy software, Software Evolution and Software Myths. Industrial Engineering Tools for Software Engineering.				8
Unit 2	**Process Models**: Software Process Models: The Waterfall Model, The Incremental Model, the RAD model, Evolution Process Model: Prototyping, The Spiral model, Concurrent Development Model. Agile Process Models: Extreme Programming (XP)				6
Unit 3	**Software Project Management:** Management Activities, Project Planning, Project scheduling, Risk management. Requirements Engineering. Feasibility study, requirement analysis, cost benefit analysis, planning systems, analysis tools and techniques.				6
Unit 4	**System Design:** design fundamentals, modular design, data and procedural design, object oriented design and UML. **System Development**: Code documentation, program design paradigms.				6
Unit 5	**Software Testing**: Test Strategies for Conventional Software, Test Strategies for Object – Oriented Software, Verification and Validation Testing, System Testing, Debugging. Black-Box and White-Box Testing, Basis Path Testing, Control Structure Testing, Regression Testing, Mutation Testing, Dataflow Testing.				8
Unit 6	**Software Maintenance**: Maintenance Characteristics, Maintainability, Maintenance Tasks and side effects				8

References

1.	Pressman Roger S, Software Engineering A Practitioner's Approach, TATA McGraw-Hill Publications, 6th Edition, 2005, ISBN No. 007-301933X
2.	Ian Sommerville, Software Engineering, Pearson Education, 7th Edition, 2008, ISBN: 978-81-7758-530-8.
3.	Ghezzi C. Jazayeri M and Mandrioli: Fundamentals of Software Engg. , PHI.
4.	Rajib Mall, Fundamentals of software engineering. PHI Learning Pvt. Ltd..
5.	Unified Modeling Language Reference manual", Grady Booch, James Rambaugh, Ivar Jacobson, Pearson India, ISBN – 9788177581614 R5.

Object Oriented Analysis and Design						
Prerequisite: : **Computer Programming skills**			L	T	P	C
Total hours: 40			3	0	0	3
Course Content						Hrs
Part I	Unit 1	Introduction to Object Oriented Programming fundamentals: Object Oriented Programming and Design, Review of abstraction Classes, Objects and Methods: Class, Object, Object reference, Constructor, Constructor Overloading				8
	Unit 2	C++ Programming Basics: Fundamentals, variables and assignments, Input and Output, Data types and expressions, flow of control, subprograms, top-down design, predefined functions, user defined functions, procedural abstractions, local variables, overloading function names, operator overloading, parameter passing, this pointer, destructors, copy constructor, overloading the assignment operator, virtual functions, function calling functions, friend functions, recursive functions, recursive member functions. Static member function.				6
	Unit 3	C++ Object oriented concepts: Objects and classes, use of file for I/O, formatting output with stream functions, Character I/O, inheritance, structures for diverse data, structures as function arguments, initializing structures, defining classes and member functions, public and private members, constructors for initialization, standard C++ classes, derived classes, flow of control, use of Boolean expressions, multiway branches, use and design of loops. Friend function and friend class				8
Part II	Unit 4	Introduction to OOD, Unified Process, UML diagrams, Use Case, Use case Modelling, Relating Use cases – include, extend and generalization – When to use Use-cases, Class Diagram, Elaboration-Domain Model, Finding conceptual classes and description classes – Associations – Attributes – Domain model refinement, Finding conceptual class Hierarchies, Aggregation and Composition – Relationship between sequence diagrams and use cases, When to use Class Diagram				6
	Unit 5	Dynamic Diagrams – UML interaction diagrams, System sequence diagram, Collaboration diagram, When to use Communication Diagrams, State machine diagram and Modelling –When to use State Diagrams, Activity diagram, When to use activity diagrams Implementation Diagrams-UML package diagram – When to use package diagrams, Component and Deployment Diagrams – When to use Component and Deployment diagrams				6
	Unit 6	Design patterns: GRASP: Designing objects with responsibilities – Creator – Information expert – Low Coupling – High Cohesion – Controller Design Patterns – creational – factory method – structural – Bridge – Adapter – behavioural – Strategy – observer –Applying GoF design patterns – Mapping design to cod				6

References	
1.	Deitel and Deitel, C++ How to Program, Third Edition, Pearson Publication.
2.	Robert Lafore, Object Oriented Programming in C++, Fourth Edition, SAMS publications.
3.	Craig Larman, ―Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development‖, Third Edition, Pearson Education, 2005.
4.	Ali Bahrami – Object Oriented Systems Development – McGraw Hill International Edition – 1999

Object Oriented Analysis and Design Lab				
Prerequisite: : Basic Computer Programming, Data Structures	L	T	P	C
Total hours: 42	0	0	2	1

Course Content	Hrs
The laboratory course will be in two parts: Part I: Object oriented Programming (OOP). In this part, students will be given OOP assignments to cover the practical on: - Programs Using Functions - Simple Classes for understanding objects, member functions and Constructors - Classes with primitive data members - Classes with arrays as data members - Classes with pointers as data members – String Class - Classes with constant data members - Classes with static member functions - Compile time Polymorphism - Operator Overloading including Unary and Binary Operators - Function Overloading - Inheritance and - Runtime Polymorphism Part II: Object Oriented Methodology and Design In this part, students will be given experiments to design various UML diagrams such as USE case, Class Diagram, State Diagram, Activity Diagram, Sequence diagram etc., based on the given case study.	30

References	
1.	Deitel and Deitel, C++ How to Program, Third Edition, Pearson Publication.
2.	Robert Lafore, Object Oriented Programming in C++, Fourth Edition, SAMS publications.
3	Craig Larman, ―Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development‖, Third Edition, Pearson Education, 2005.
4	Ali Bahrami – Object Oriented Systems Development – McGraw Hill International Edition – 1999