Tentative UG(AI&DE) Scheme

Department of Computer Science and Engineering

First Semester					
S. No	**Code**	**Subject**	**L-T-P**	**Credit**	**Type**
		Programming with Python	***2-0-0***	***2***	***IC***
		Programming lab	***0-0-2***	***1***	***IC***
		Other Institute Core Subjects		***15***	***IC***
	CST1xx	Problem Solving using C	2-0-0	2	DC
	AIT1xx	Discrete Mathematics	3-0-0	3	DC
	AIP1xx	Problem Solving Using C Lab	0-0-2	1	DC
				24	

Second Semester					
S. No	**Code**	**Subject**	**L-T-P**	**Credit**	**Type**
		Programming with Python	***2-0-0***	***2***	***IC***
		Programming with Python lab	***0-0-2***	***1***	***IC***
		Other Institute Core Subjects		***15***	***IC***
	AIT1xx	Data Structures	3-0-0	3	DC
	AIT1xx	Foundation of Learning	3-0-0	2	DC
	AIP1xx	Data Structures Lab	0-0-4	2	DC
				25	

Third Semester					
S. No	**Code**	**Subject**	**L-T-P**	**Credits**	**Type**
	AIT2xx	Digital Systems and Computer Architecture	4-0-0	4	DC
	AIT2xx	Design and Analysis of Algorithms	3-0-0	3	DC
	AIT2xx	Artificial Intelligence	3-0-0	3	DC
	AIT2xx	Foundations of data science	3-1-0	4	DC
	AIT2xx	Theory of Computation	3-0-0	3	DC

	MMT2xx	Social Sciences and Professional Ethics	2-1-0	3	BS
	AIP2xx	Digital Systems Lab	0-0-2	1	DC
	AIP2xx	Design and Analysis of Algorithms Lab	0-0-4	2	DC
	AIP2xx	Artificial Intelligence Lab	0-0-4	2	DC
				25	

Fourth Semester					
S. No	**Code**	**Subject**	**L-T-P**	**Credits**	**Type**
	AIT2xx	Artificial Neural Networks	3-0-0	3	DC
	AIT2xx	Operating Systems	3-0-0	3	DC
	AIT2xx	Compiler	3-0-0.	3	DC
	AIT2xx	Machine Learning	3-0-0	3	DC
	AIT2xx	Database Information Systems	3-0-0	3	DC
	MMT2xx	Basics of Managements	3-0-0	3	MM
	AIT2xx	Technical Writing	1-0-2	2	DC
	AIP2xx	Machine Learning Lab	0-0-2	1	DC
	AIP2xx	Operating System Lab	0-0-4	2	DC
	AIP2xx	Database Information Systems Lab	0-0-4	2	DC
				25	

Fifth Semester					
S. No	**Code**	**Subject**	**L-T-P**	**Credits**	**Type**
	AIT3xx	Digital Image Processing	3-0-0	3	DC
	AIT3xx	Software Engineering	3-0-0	3	DC
	AIT3xx	Computer Networks	3-0-0	3	DC
	AIT3xx	Data Analytics	3-0-0	3	DC
	AIT3xx		3-0-0	3	DC/PLEAS
	AIT3xx	Program Elective-1	3-0-0	3	PE
	AIP3xx	Operating System Lab	0-0-2	1	DC
	AIP3xx	Compiler Design Lab	0-0-2	1	DC
	AIP3xx	Computer Networks Lab	0-0-4	2	DC
				22	

Honors					
	AITxxx	Advance Data Structures and Algorithms		3	
	AITxxx	Machine Learning		3	
				6	

Minor AIDE					
	AITxxx	Data Structures		3	OE
	AITxxx	Operating System		3	DC
				6	

Sixth Semester					
S. No	**Code**	**Subject**	**L-T-P**	**Credits**	**Type**
	AIT3xx	Deep Learning	3-0-0	3	DC
	AIT3xx	Natural Language Processing	3-0-0	3	DC
	AIT3xx	IOT and Robotics	3-0-0	3	DC
	AIT3xx	Program Elective-2	3-0-0	3	PE
	EExxx	Smart Grid	3-0-0	3	PLEAS
	AIT3xx	Computer and Network Security	3-0-0	3	DC
	AIP3xx	Deep Learning Lab	0-0-4	2	DC
	AIP3xx	Natural Language Lab	0-0-2	1	DC
	AIP3xx	IOT and Robotics Lab	0-0-4	2	DC
				23	

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

Minor AIDE					
	AITxxx	Computer Networks		3	DC
	AITxxx	Database Information Systems		3	DC
				6	

Seventh Semester					
S. No	**Code**	**Subject**	**L-T-P**	**Credits**	**Type**
1		Open Elective – 1	3-0-0	3	OE
2		Minor Project		3	DC
	AITxxx	Advance Elective-1	3-0-3	5	AE
	AITxxx	Advance Elective-2	3-0-3	5	AE
	AIP7xx	Training Seminar	0-0-3	2	DC
				18	
Honors					
	AITxxx	Honors Elective-3		3	
				3	

Minor AIDE					
	AITxxx	Artificial Intelligence		3	DC
				3	

Eighth Semester					
S. No	**Code**	**Subject**	**L-T-P**	**Credits**	**Type**
1		Open Elective – 2	3-0-0	3	OE
2		Major Project	0-0-12	6	
	AITxxx	Advance Elective-3	3-0-3	5	AE
	AITxxx	Advance Elective-4	3-0-3	5	AE
				19	

Honors					
	AITxxx	Honors Elective-4		3	
				3	

Minor AIDE					
	AITxxx	Federated Learning		3	DC
				3	

Programming with Python					
Prerequisite: :NiL		L	T	P	C
Total hours: 28		2	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 like matplotlib, numpy, 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.SojanLal, "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, Orelly 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 Beginers" , Third Edition, Course Technology Cengage Learning Publications, 2013, ISBN 978-1435455009
8.	David Beazley, Brian Jones., "Python Cookbook", Third Edition, Orelly Publication, 2013, ISBN 978-1449340377

Problem solving using C					
Prerequisite: :NiL		L	T	P	C
Total hours: 28		2	0	2	3

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	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, Fibonaaci through storing previous values – introduction to dynamic programming, Nested loops through sorting methods; use of break and continue Bitvector implementation of set and usage of bitwise operators for testing membership (withing set), union and intersection of two sets Macro & Preprocessor in C	

Unit 4	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	6
Unit 5	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.	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

Discrete Mathematics					
Prerequisite: :NiL		L	T	P	C
Total hours: 42		3	0	0	3
Course Content					Hrs
Unit 1	Logic: Truth Tables, Conditionals ($P \Rightarrow Q$), and Bi-conditionals ($P \Leftrightarrow Q$), Negation, Converse, and Contrapositive, Existential and Universal Quantifiers ($\forall, \exists, \exists!$), 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

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, 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.

	Course Title:	Foundations of Learning
	Course Hours:	**L** **3** **T** **1** **P** **0**
	Credits:	**4**
	Prerequisites:	Some basic set theory (what is a set and elementary set operations), combinatory (knowing different ways of counting, inclusion-exclusion principle) and calculus (knowing derivatives and integrals)
	Course Outcomes:	
	1.	This course introduces the student to various fundamental concepts in probability theory and linear algebra.
	2.	The knowledge of such mathematical tools is essential and provides a foundation for various fields of computer science like Machine Learning, Communication Networks, Computer Graphics and Vision etc.
	3.	Though the treatment of the subject is mathematical, focus is more on the problem solving techniques rather than on the formalism.
	Course Contents:	
	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, Eigendecomposition, Singular Value Decomposition, The Moore-Penrose Pseudoinverse, The Trace Operator, The Determinant, Principal Component Analysis 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 Useful Properties of Common Functions, Technical Details of Continuous Variables, Information Theory, Structured Probabilistic Models Statistical inference: statistical decision theory, statistical assumptions, estimation theory. Methods of estimation: method of moments, method of minimum variance 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.	
	Suggested Books	
	1.	Linear Algebra, Gilbert Strang, MIT Cambridge Press
	2.	Chapter 3, Deeplearning, Ian Goodfellow, MIT Cambridge Press
	3.	Probability and Statistics for Machine Learning, Anirban Das Gupta, Springer
	4.	The Elements of Statistical Learning, second ed, Springer

Data Structures Lab				
Prerequisite: :NiL	L	T	P	C
	0	0	4	2

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,2ndEdition,Pearson.

Digital Systems and Computer Architecture					
Prerequisite:		L	T	P	C
***Total* Hours: 42**		**3**	**0**	**0**	**3**

Course Content		Hrs
Unit 1	Number Systems and Codes: Representation of Negative. Numbers; 1's Complement and 2's Complement, Complement Arithmetic, BCD Arithmetic, Digital Codes -Excess-3 code, Gray code, Binary to Excess- code conversion and vice versa, ASCII code, EBCIDIC code, Error Detection Codes. Logic Gates, Universal Gates and their characteristic: K-Map, SOP, POS.	8
Unit 2	Combinational circuits: Adders, Subtractors, Binary Parallel Adder – Carry look ahead Adder, BCD Adder, Multiplexer, Demultiplexer, Comparator, Decoder and Encoder. Sequential Circuits: Latches, Flip-Flops: RS, D Type, JK, and T Type and their conversion, Master-Salve Flip and Race Conditions. Registers: Design of shift registers and their operations. Counters: Asynchronous and Synchronous counters, Applications of counters.	8
Unit 3	Introduction to computer architecture: Digital components, Von Neumann Machine Architecture, Flynn Classification Register Transfer Language: Micro operations - data transfer operations, arithmetic, logic and shift micro operations and their hardware implementations as a simple Arithmetic and logic unit.	9
Unit 4	CPU Organization: Addressing techniques - Immediate, direct, indirect, register, register indirect, index, relative and stack addressing techniques, Instruction formats, Instruction set design, Instruction types	9
Unit 5	Arithmetic Algorithms: Arithmetic and Logic Unit, Adders - Full adder, Ripple carry adder, Carry look ahead adder, Carry select adder, carry save adder, Multiplication - Add and Shift method, Booth's Multiplier, m -Array Multiplier, Division - Restoring and Non restoring method.	8
References		

1.	Herbert Taub, Donald L. Schilling, Digital Integrated Electronics, McGraw-Hill,
2.	M. Morris Mano, Digital Logic and Computer Design, Person Education
3.	John P. Hayes, “Computer Architecture and Organization”, Tata McGraw Hill, Third Edition
4.	William Stallings, “Computer Organization and Architecture – Designing for Performance”, Pearson Education, Seventh Edition, 2006.

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.	8

	NP-Complete Examples with Reductions: Satisfiability, Clique, Independent Set, Vertex Cover , Graph Coloring, Dominating Set,	

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.

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 i. What is AI ii. Foundation of AI and its history iii. Agents and Environment	8
Unit 2	Problem Solving i. Solving problem by searching ii. Beyond classical search iii. Adversarial search iv. Constraint satisfaction problems	8
Unit 3	Knowledge, reasoning and planning i. Logical agents ii. First order logic iii. Inference in First order logic iv. Knowledge representation	9
Unit 4	Uncertain knowledge and reasoning i. Quantifying uncertainity ii. Probabilistic reasoning iii. Probabilistic reasoning overtime iv. Inference in temporal models v. Hidden markov models vi. The basis of utility theory vii. Utility functions\ viii. Multiattribute utility functions	9
Unit 5	Learning i. Learning from examples ii. Evaluating and choosing the best hypothesis iii. The theory of learning iv. Knowledge in learning	8

References

1.	Artificial Intelligence a Modern Approach, III Edition, Stuart Russell and Peter Norvig

2.	Probability and Statistics for Machine Learning, Anirban Das Gupta, Springer
3.	The Elements of Statistical Learning, Trevor Hastie, Robert Tibshirani, second ed, Springer

Foundation of Data Science					
Prerequisite: Foundation of Learning		L	T	P	C
Total Hours: 40		***2***	***0***	***0***	***2***
Course Content					Hrs
Unit 1	Roles in a Data Science project, Setting expectations, Data Science methodology , Business understanding, Data Requirements, Data Acquisition, Data Understanding, Data preparation, Modelling, Model Evaluation, Deployment and feedback, Data Science Process, Roles in a Data Science project				10
Unit 2	About Data- Data quality, Data representation, Data Models, Data Sampling, Data Visualization: Basic principles, ideas and tools for data visualization. Data Wrangling- Feature Engineering, Feature Selection				10
Unit 3	Data preprocessing: Data cleaning – data integration – Data Reduction Data Transformation and Data Discretization. Evaluation of classification methods – Confusion matrix, Students T-tests and ROC curves-Exploratory Data Analysis – Basic tools (plots, graphs and summary statistics) of EDA, Philosophy of EDA – The Data Science Process.				10
Unit 4	Ethics for Data Science- Ethical guidelines for Data Scientist, Societal consequences, Ethics of data scraping and storage, Rightful use of data science				10
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.				

Social Sciences and Professional Ethics Course code: HST202				
Prerequisite: Nil	L	T	P	C
Total hours: 42	2	1	0	3

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	8
Unit 2	**Social Change** Social Change, development and progress; Globalisation, Industrialisation, urbanisation and modernisation; Social mobility and social stratification	6
Unit 3	**Social Issues** Science technology and society; Digital divide, Appropriate technology, Gender inequality; Substance abuse, Consumerism, Environmental degradation and climate crisis, Nation building	8
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.	8
Unit 5	**Finance and banking**	6

	Banking and Financial Sector; Reforms & Challenges; Monetary & Fiscal Policies; meaning, importance & instruments. Global economic environment and opportunities. Intellectual property rights and R & D environment.	
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	6

References

1.	Haralambos and Holborn: Sociology: Themes and Perspective.
2.	G, Ritzer: Sociological Theories
3.	William Lillie, An introduction to Ethics/"Ethics for the New Millennium" by the Dalai Lama
4.	Uma Kapila, Indian Economy Performance and Policies (Latest Edition), Academic Foundation, New Delhi
5.	Ahluwalia, I.J. & IMD Little, India’s Economic Reform and Development, Oxford University Press, India

Theory of Computation					
Prerequisite: NIL		L	T	P	C
Total hours: 40		3	1	0	4
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,				8

	Multitrack, Mutitape, Nondeterministic, TM, Encoding of TM, Computability and Acceptability.	
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	K. L. P Mishra, N. Chandrashekaran, Theory of Computer Science-Automata Languages and Computation, Prentice Hall of India, India.

Design and Analysis of Algorithms Lab				
Pre-requisite: C Programming, Data Structures	L	T	P	C
	0	0	3	2

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.

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.

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.

Operating System				
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	***<u>Introduction:</u>*** What is an operating system, Types of operating systems and differences among them, 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. ***<u>CPU Scheduling:</u>*** 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	***<u>Synchronization:</u>*** 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	***<u>I/O Management:</u>*** 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 ***<u>File management:</u>*** 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	***<u>Security and Protection:</u>*** 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.

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.

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

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.

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.

Technical Writing				
Prerequisite: :NiL	L	T	P	C
Total hours:	1	0	2	2
Course Content				Hrs

	Introduction to Documentation using Doxygen, Google Docs, Latex/ Overleaf Drawing software (e.g. inkscape, xfig, open-office) Presentation using Beamer:Introduction to creating slides, adding frames, dividing the slide into multiple columns, adding different blocks, etc.. Graph plotting software (e.g., gnuplot) Version control tools - GIT /GitHub/SVN	
	Introduction: LaTeX, its installation, and different IDEs. The learner creates the first document using LaTeX, organizes content into sections using article and book class of LaTeX. Styling Pages: Reviewing different paper sizes, examines packages, formats the page by setting margins, customizing header and footer, changing the page orientation, dividing the document into multiple columns. Different types of error messages. Formatting Content: formatting text (styles, size, alignment), adding colors to text and entire page, and adding bullets and numbered items, the process of writing complex mathematics. Tables and Images: creating basic tables, adding simple and dashed borders, merging rows and columns, and handling situations where a table exceeds the size of a page. Add an image, explore different properties like rotate, scale, etc.. Referencing and Indexing: the learner learns to add cross-referencing (refer to sections, table, images), add bibliography (references), and create back index.	

References

1.	Latex - A document preparation system, 2/e, by Leslie Lamport, Addison-Wesley, 1994
2.	https://www.doxygen.nl/

<table>
<tr><td colspan="5">Machine Learning Lab</td></tr>
<tr><td colspan="2">Prerequisite: : Python Programming</td><td>L</td><td>T</td><td>P</td><td>C</td></tr>
<tr><td colspan="2">Total hours: 42</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="5">Course Content</td><td>Hrs</td></tr>
<tr><td>1</td><td colspan="4">Perceptron Learning Algorithm:

1. Generate a linearly separable data (random) set of size 20. Plot the examples $\{(x_n, y_n)\}$ as well as the target function f on a plane. Be sure to mark the examples from different classes differently, and add labels to the axes of the plot.

2. Run the perceptron learning algorithm on the data set above. Report the number of updates that the algorithm takes before converging. Plot the examples $\{(x_n, y_n)\}$, the target function f , and the final hypothesis g in the same figure. Comment on whether f is close to g.

Repeat everything in (2) with another randomly generated data set of size 100. Compare your results with (2)</td><td>3</td></tr>
<tr><td>2</td><td colspan="4">Linear Regression:

Write a python script that can find w0 and w1 for an arbitrary dataset of number of hours studied versus rank of a students as $\{(x_n, y_n)\}$ pairs. Find the linear model, y = w T x, that minimizes the squared loss. Derive the optimal w for the total training loss:s MSE/RSS $L = \Sigma(yn - wT\ xn)^2$. Using the model predict the rank for the number of hours studied.

Load the data stored in the file syntheticdata.mat. Fit a 4th order polynomial function $f(x; w) = w_0 + w_1 x + w_2 x^2 + w_3 x^3 + w_4 x^4$ to this data. What do you notice about w2 and w4 ?)Fit a function $f(x; w) = w_0 + w_1 x + w_2 \sin((x-a)'/b)$, assuming a and b are fixed in some sensible range. Show a least square fit using this model. What do you notice about w1 and w2 . Comment about generalization and overfitting.</td><td>3</td></tr>
<tr><td>3</td><td colspan="4">Logistic Regression: Handwritten Digits Data: You should download the two data files with handwritten digits data: training data (ZipDigits.train) and test data (ZipDigits.test). Each row is a data example. The first entry is the digit, and the next 256 are grayscale values between −1 and 1. The 256 pixels correspond to a 16 × 16 image. For this problem, we will only use the 1 and 4 digits, so remove the other digits from your training and test examples. Please submit your Python code implementing the logistic regression for classification using gradient descent. Familiarize yourself with the data by giving a plot of two of the digit images. Develop two features to measure properties of the image that would be useful in distinguishing between 1 and 4. You may use symmetry and average intensity (as discussed in class). As in the text, give a 2-D scatter plot of your features: for each data example, plot the two features with a red redx if it is a 4 and a blue blueo if it is Classifying Handwritten Digits: 1 vs. 4. Implement logistic regression

for classification using gradient descent to find the best separator you can using

the training data only (use your 2 features from the above question as the inputs). The output is +1 if the example is a 1 and −1 for a 4. Give separate plots of the training and</td><td>6</td></tr>
</table>

	test data, together with the separators. Compute E in on your training data and E test , the test error on the test data after 1000 iterations. Now repeat the above using a 3 rd order polynomial transform. As your final deliverable to a customer, would you use the linear model with or without the 3 rd order polynomial transform? Explain. **Regularization:** Logistic regression can also be augmented with the l_2 -norm regularization: min $E(w) + \lambda\|W\|_2^2$, where E(w) is the logistic loss. Please change your gradient descent algorithm accordingly and use cross-validation to determine the best regularization parameter. Plot the training and testing performance curves.Indicate in the plot the best regularization parameter you obtained (using cross validation).	
4	**Neural Networks:** In this problem you will implement forward and backward propagation methods for a multi-layer neural network with K hidden layers. Assume that K is a user input less than 10. Implement the networks separately with the following activation functions: • Sigmoid: Derive the gradient of the activation function. Confirm with numerical differentiation. . Tanh: Derive the gradient of the activation function. Confirm with numerical differentiation. Assume that the last layer has a linear activation function and the loss function is $l(y, \hat{y}) = \|y - \hat{y}\|_2^2$. Submit your code (along with any instructions necessary to # run it), the forward pass outputs at each layer and the gradients of the parameters (W_{ij}^k, b_i^k) . The input, output and the parameters of the network can be found in the MAT file associated with this problem. In this problem you will train a multi-layer neural network to recognize handwritten digits. Use the multi-layer neural network (with ReLU activation) that you implemented in the previous homework. Use 32 nodes in each layer and initialize the weights randomly. The data is also provided to you in a MAT file. • Report the training and validation accuracy as a function of iterations (with 5 hidden layers). Report the convergence speed of the training procedure (with 5 hidden layers) for the Stochastic Gradient Descent optimization algorithm. • Determine the number of hidden layers required via cross-validation. Report the training and validation accuracy for cross-validation. • Finally, report the best test error that you can achieve.	6
5	**Evaluation Metrics:** Consider a theoretical biometric matcher that generates distance scores in the range $[-\infty,\infty]$. Assume that the genuine and impostor score distributions due to this matcher can be approximately modeled as N(30, 10) and N(60, 15), respectively. Here, $N(\mu, \sigma 2)$ denotes normal distribution with mean, μ, and variance, $\sigma 2$ Suppose the following decision rule is employed: s is classified as a genuine score if $s \leq \eta$; else it is classified as an impostor score. Here, $\eta \in [0, 100]$.	4

	• Plot the genuine and impostor distributions in a single graph. The distributions should be contained in the range [0, 100]. • If η = 50, what is the FMR (i.e., FAR) and FNMR (i.e., FRR) of the biometric matcher? • Given s is classified as a genuine score if s ≤ η; else it is classified as an impostor score. If η = 75, what is the FMR (i.e., FAR) and FNMR (i.e., FRR) of the biometric matcher? • Plot the DET curve of this matcher. • Plot the ROC curve and AUC of this matcher.	
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 three tables: Table one with attributes: Id, Exercise, Family history, Heart Attack Risk. Table two with attributes: Id, Smoker, Obese, Heart Attack Risk, Table three: Id, Obese, Falimy history and Heart Attack Risk. Generate s100 samples randomly for the three tables. List three bootstrap samples, using these bootstrap samples create decision trees that will be in the random forest model using entropy based information gain as the feature selection criteria. Assuming the random forest uses majority voting, what prediction will it return for the query: EXERCISE = rarely, SMOKER = false, OBESE = true, FAMILY = yes.	6
8	**Clustering:** A bank wants to detect fradulent credit card transactions. Using random function generate data for lots of transactions (each transaction is an amount of money, a shop, and the time and date) and some information about which credit cards were stolen, and the transactions that were performed on the stolen card. Generate random data files for the above description of atleast 200 transactions. Implement Agglomerative, Hierarchichal and Density based clustering techniques to cluster people's transactions together to identify patterns, so that stolen cards can be detected as changes in pattern. How well do you think this will work? There is much more data of transactions when cards are not stolen, compared to stolen transactions. How does it affect the learning, and what can you do about it.	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

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.				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.

Data Analytics					
Prerequisite: :NiL		L	T	P	C
Total hours: 35		0	0	3	2
Course Content					Hrs
1	Data Science Overview, Evolution of Data Science, Tools for Data Science, Applications of Data Science, Retrieving Data, Data Preparation, Data Exploration, Data Modelling, Numerical Operations on Arrays, Array Functions, Data Processing using Arrays, Loading and Saving Data, Saving an Array, Loading an Array, Numpy Random Numbers Data Manipulation with Pandas: Data Wrangling, Data Exploration, Cleaning Data, Filtering, Merging Data, Reshaping Data, Data Aggregation, Reading and Writing Files, Loading and Saving Data with Pandas.				6
2	Data Visualization with Python, Data Visualization, Bar Charts, Line Plot, Area Plots, Histograms, Pie Charts, Box Plots, Scatter Plots, Time Series plots, Figures and Subplots, Plotting Functions with Pandas. Data Visualization using non programming tools like Tableau. Work with Filter, Parameters, Sets. Arithmetic and logical table. Data visualization techniques such as heat map, tree map, Pareto.				6
3	Fundamentals of R, Basic Statistics in R, Data Cleaning & Visualization in R, Linear Regression in R, Logistic Regression in R, Segmentation for marketing analytics in R, Time series forecasting in R, Decision Trees in R, Random Forest & XGBoost in R, Solving an actual business problem through analytics				9
4	Overview of Database Management Systems, Introduction to Big Data, Introduction to distributed file system, Big Data and its importance, Four Vs, Drivers for Big data, Big data analytics. Apache Hadoop & Hadoop Eco-System, Moving Data in and out of Hadoop, Understanding inputs and outputs of MapReduce, Data Serialization.				3

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.

<table>
<tr><th colspan="6">Digital Image Processing</th></tr>
<tr><td colspan="2">Prerequisite: Fundamental knowledge on signals and systems, basics of linear algebra and calculus, and programming skills</td><td>L</td><td>T</td><td>P</td><td>C</td></tr>
<tr><td colspan="2">Total hours: 42</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 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</td><td>6</td></tr>
<tr><td>Unit 2</td><td colspan="4">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</td><td>8</td></tr>
<tr><td>Unit 3</td><td colspan="4">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</td><td>8</td></tr>
<tr><td>Unit 4</td><td colspan="4">Image Restoration: Overview of Degradation models, Unconstrained and constrained restorations, Inverse Filtering, Wiener Filter</td><td>6</td></tr>
<tr><td>Unit 5</td><td colspan="4">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</td><td>8</td></tr>
<tr><td>Unit 6</td><td colspan="4">Representation and Description: Representation schemes, boundary descriptors, regional descriptors.
Morphology: Dilation, erosion, opening, closing, Hit-or-Miss Transform, some basic morphological algorithms</td><td>6</td></tr>
<tr><td colspan="6">References</td></tr>
<tr><td>1.</td><td colspan="5">Rafael C. Gonzalez, Richard E. Woods, Digital Image Processing, Pearson , 3rd Edition, 2008</td></tr>
<tr><td>2.</td><td colspan="5">Castleman. Digital Image Processing. Prentice Hall.</td></tr>
<tr><td>3.</td><td colspan="5">Anil K. Jain, Fundamentals of Digital Image Processing, Pearson , 2002</td></tr>
</table>

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

Digital Image Processing Lab				
Prerequisite: Fundamental knowledge on image processing and programming skills	L	T	P	C
	0	0	2	1

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

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	36

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

Data Analytics Lab

Prerequisite: :NiL	L	T	P	C
Total hours: 35	0	0	3	2

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

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	3	5
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,				10

	Question Answering Systems, Social network analysis, Text Summarization.	

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.

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.

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
	1. Familiarization of cloud based computing like Google colab 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.				