Scheme & Syllabus of

Bachelor of Technology

Artificial Intelligence & Machine Learning

Batch 2020 onwards
(3rd -8th Semester)

By

Department of Academics

IK Gujral Punjab Technical University

Bachelor of Technology in AI & ML

It is a Graduate (UG) Programme of 4 years duration (8 semesters)

Courses & Examination

Scheme: Third Semester

Course Code	Type of Course	Course Title	Hours per Week			Marks Distribution		Total Marks	Credits
			L	T	P	Internal	External		
BTES 301-18	Engineering Science Course	Digital Electronics	3	0	0	40	60	100	3
BTCS 301-18	Professional Core Courses	Data structure & Algorithms	3	0	0	40	60	100	3
BTCS 302-18	Professional Core Courses	Object Oriented Programming	3	0	0	40	60	100	3
BTAM 304-18	Basic Science Course	Mathematics-III	3	0	0	40	60	100	3
HSMC 101/102-18	Humanities & Social Sciences Including Management \Courses	Foundation Course in Humanities (Development of Societies/Philosophy)	2	1	0	40	60	100	3
BTES 302-18	Engineering Science Course	Digital Electronics Lab	0	0	2	30	20	50	1
BTCS 303-18	Professional Core Courses	Data structure & Algorithms Lab	0	0	4	30	20	50	2
BTCS 304-18	Professional Core Courses	Object Oriented Programming lab.	0	0	4	30	20	50	2
BTCS 305-18	Professional Core Courses	IT Workshop*	0	0	2	30	20	50	1
		Summer Institutional Training	0	0	0	0	0	0	Satisfactory/Unsatisfactory
Total			**14**	**1**	**12**	**320**	**380**	**700**	**21**

*Syllabus to be decided by respective institute internally. It may include latest technologies.

Fourth Semester

Course Code	Type of Course	Course Title	Hours per Week			Marks Distribution		Total Marks	Credits
			L	T	P	Internal	External		
BTCS 401-18	Professional Core Courses	Discrete Mathematics	3	1	0	40	60	100	4
BTES 401-18	Engineering Science Course	Computer Organization & Architecture	3	0	0	40	60	100	3
BTCS 402-18	Professional Core Courses	Operating Systems	3	0	0	40	60	100	3
BTCS 403-18	Professional Core Courses	Design & Analysis of Algorithms	3	0	0	40	60	100	3
HSMC 122-18	Humanities & Social Sciences including Management Courses	Universal Human Values 2	2	1	0	40	60	100	3
EVS101-18	Mandatory Courses	Environmental Sciences	3	-	-	100	-	100	S/US
BTES 402-18	Engineering Science Course	Computer Organization & Architecture Lab	0	0	2	30	20	50	1
BTCS 404-18	Professional Core Courses	Operating Systems Lab	0	0	4	30	20	50	2
BTCS 405-18	Professional Core Courses	Design & Analysis of Algorithms Lab	0	0	4	30	20	50	2
Total			**15**	**2**	**10**	**390**	**360**	**750**	**24**

Students will take up summer internship of 4-6 weeks at industry or organizations of repute after 4th sem, that will be accredited in 5th semester.

Fifth Semester

Course Code	Type of Course	Course Title	Hours per Week			Marks Distribution		Total Marks	Credits
			L	T	P	Internal	External		
BTES 501-20	Engineering Science	Statistical Computing Techniques using R	3	0	0	40	60	100	3
BTCS 501-18	Professional Core Courses	Database Management Systems	3	0	0	40	60	100	3
BTCS 502-18	Professional Core Courses	Formal Language & Automata Theory	3	0	0	40	60	100	3
BTAIML 501-20	Professional Core Courses	Programming in Python	3	0	0	40	60	100	3
BTAIML 502-20	Professional Core Courses	Artificial Intelligence	3	0	0	40	60	100	3
BTAIML ***	Professional Elective	Elective-I	3	0	0	40	60	100	3
MC	Mandatory Courses	Constitution of India/ Essence of Indian Traditional Knowledge	2	-	-	100	-	100	S/US
BTES 502-20	Engineering Science	Statistical Computing Techniques using R lab	0	0	2	30	20	50	1
BTCS 505-18	Professional Core Courses	Database Management Systems lab	0	0	2	30	20	50	1
BTAIML 503-20	Professional Core Courses	Programming in Python Lab	0	0	2	30	20	50	1
BTAIML 504-20	Professional Core Courses	Artificial Intelligence Lab	0	0	2	30	20	50	1
BTAIML ***	Professional Elective	Elective-I Lab	0	0	2	30	20	50	1
	Professional Training	Industrial *Training	-	-	-	60	40	100	S/US
Total			**20**	**0**	**10**	**460**	**440**	**900**	**23**

* 4-6 weeks industrial training undertaken after 4th semester in summer vacations.

Sixth Semester

Course Code	Type of Course	Course Title	Hours per Week			Marks Distribution		Total Marks	Credits
			L	T	P	Internal	External		
BTCS 504-18	Professional Core Courses	Computer Networks	3	0	0	40	60	100	3
BTCS 619-18	Professional Core Courses	Machine Learning	3	0	0	40	60	100	3
BT* UUU-18	Professional Elective Courses	Elective-II	3	0	0	40	60	100	3
BT* YYY-18	Professional Elective Courses	Elective-III	3	0	0	40	60	100	3
BTOE ***	Open Elective Courses	Open Elective-I	3	0	0	40	60	100	3
BTCS 603-18	Project	Project-1	0	0	6	60	40	100	3
BTCS 507-18	Professional Core Courses	Computer Networks Lab	0	0	2	30	20	50	1
BTCS 620-18	Professional Core Courses	Machine Learning Lab	0	0	2	30	20	50	1
BT* UUU-18	Professional Elective Courses	Elective-II lab	0	0	2	30	20	50	1
BT* YYY-18	Professional Elective Courses	Elective-III lab	0	0	2	30	20	50	1
Total			**15**	**0**	**14**	**380**	**420**	**800**	**22**

Seventh/ Eighth Semester

Course Code	Type of Course	Course Title	Hours per Week			Marks Distribution		Total Marks	Credits
			L	T	P	Internal	External		
BTCS 601-18	Professional Core Courses	Compiler Design	3	0	0	40	60	100	3
BTAIML709-20	Professional Core Courses	Applied Intelligence	3	0	0	40	60	100	3
BTOE ***	Open Elective Courses	Open Elective-II	3	0	0	40	60	100	3
BT* ZZZ-18	Professional Elective	Elective- IV	3	0	0	40	60	100	3
BT* TTT-18	Professional Elective Courses	Elective-V	3	0	0	40	60	100	3
BTCS 703-18	Project	Project-II	0	0	12	120	80	200	6
BT* ZZZ-18	Professional Elective	Elective- IV lab	0	0	2	30	20	50	1
BT* TTT-18	Professional Elective	Elective- V lab	0	0	2	30	20	50	1
Total			**15**	**0**	**16**	**380**	**420**	**800**	**23**

Seventh/Eighth Semester

Course Code	Course Title	Marks Distribution		Total Marks	Credits
		Internal	External		
BTCS 801-20	Semester Training	300	200	500	16

LIST OF ELECTIVES

Elective I

BTAIML 505-20 Data Visualization using tableau
BTAIML 506-20 Data Visualization using tableau lab
BTAIML 507-20 User Interface development
BTAIML 508-20 User Interface development lab
BTAIML 509-20 Java Programming
BTAIML 510-20 Java Programming lab

Elective II

BTCS702-18 Data Mining and Data Warehouse
BTAIML609-20 Data Mining and Data Warehouse Lab
BTAIML601-20 Graph Theory
BTAIML602-20 Graph Theory Lab
BTDS603-20 Big Data Analytics
BTDS604-20 Big Data Analytics Lab

Elective III

BTAIML603-20 Neural Networks
BTAIML604-20 Neural Networks Lab
BTAIML605-20 Recommender System
BTAIML606-20 Recommender System Lab
BTAIML607-20 Advance Computing and Network Technologies
BTAIML608-20 Advance Computing and Network Technologies Lab

Elective IV

BTAIML703-20 NLP and Information Retrieval
BTAIML704-20 NLP and Information Retrieval Lab
BTAIML705-20 Network Security Applications using AI
BTAIML706-20 Network Security Applications using AI Lab
BTAIML707-20 Robotics and Intelligent systems
BTAIML708-20 Robotics and Intelligent systems Lab

Elective V

BTCS704-18 Deep Learning
BTCS705-18 Deep Learning Lab
BTAIML701-20 Computer Vision
BTAIML713-20 Computer Vision Lab
BTAIML711-20 Augmented and Virtual Reality
BTAIML712-20 Augmented and Virtual Reality Lab

Third Semester

IK Gujral Punjab Technical University, Kapurthala
B. Tech, Computetr Science & Engg. with AI & ML

Course Code: BTCS301-18	**Course Title:** Data Structure & Algorithms	**3L:0T:P**	**3Credits**

Detailed Contents:

Module 1: Introduction

Basic Terminologies: Elementary Data Organizations, Data Structure Operations: insertion, deletion, traversal etc.; Analysis of an Algorithm, Asymptotic Notations, Time-Space trade off.

Searching: Linear Search and Binary Search Techniques and their complexity analysis.

[6 hrs] (CO1)

Module 2: Stacks and Queues

ADT Stack and its operations: Algorithms and their complexity analysis, Applications of Stacks: Expression Conversion and evaluation – corresponding algorithms and complexity analysis. ADT queue, Types of Queue: Simple Queue, Circular Queue, Priority Queue; Operations on each types of Queues: Algorithms and their analysis.

[10 hrs] (CO2, CO4, CO5)

Module 3: **Linked Lists**

Singly linked lists: Representation in memory, Algorithms of several operations: Traversing, Searching, Insertion into, Deletion from linked list; Linked representation of Stack and Queue, Header nodes, Doubly linked list: operations on it and algorithmic analysis; Circular Linked Lists: All operations their algorithms and the complexity analysis.

Trees: Basic Tree Terminologies, Different types of Trees: Binary Tree, Threaded Binary Tree, Binary Search Tree, AVL Tree; Tree operations on each of the trees and their algorithms with complexity analysis. Applications of Binary Trees. B Tree, B+ Tree: definitions, algorithms and analysis.

[10 hrs] (CO2, CO4, CO5)

Module 4: Sorting and Hashing

Objective and properties of different sorting algorithms: Selection Sort, Bubble Sort, Insertion Sort, Quick Sort, Merge Sort, Heap Sort; Performance and Comparison among all the methods, Hashing.

[10 hrs] (CO3)

Module 4: Graph

Basic Terminologies and Representations, Graph search and traversal algorithms and complexity analysis.

[6 hrs] (CO2, CO4)

Course Outcomes:

The student will be able to:

1. For a given algorithm student will able to analyze the algorithms to determine the time and computation complexity and justify the correctness;
2. Student will be able to handle operation like searching, insertion, deletion, traversing on various Data Structures and determine time and computational complexity;
3. Student will able to write an algorithm Selection Sort, Bubble Sort, Insertion Sort, Quick Sort, Merge Sort, Heap Sort and compare their performance in term of Space and Time complexity;
4. Students will be able to choose appropriate Data Structure as applied to specific problem definition; &

5. Demonstrate the reusability of Data Structures for implementing complex iterative problems.

Suggested Books:

1. "Classic Data Structures", Samanta and Debasis, 2nd edition, PHI publishers.
2. "Fundamentals of Data Structures", Illustrated Edition by Ellis Horowitz, SartajSahni, Computer Science Press.
3. "Data Structures with C (Schaum's Outline Series)", Seymour Lipschutz, 1st edition,McGraw Hill Education.

Reference Books:

1. Algorithms, Data Structures, and Problem Solving with C++", Illustrated Edition by Mark Allen Weiss, Addison-Wesley Publishing Company.
2. "How to Solve it by Computer", 2nd Impression by R. G. Dromey, Pearson Education.

Course Code: BTCS302-18	**Course Title:** Object Oriented Programming	**3L:0T:0P**	**3Credits**

Pre-requisites: Programming in C

Detailed Contents:

Module 1: Introduction

Overview of C++, Sample C++ program, Different data types, operators, expressions, and statements, arrays and strings, pointers & function components, recursive functions, user - defined types, function overloading, inline functions, Classes & Objects – I: classes, Scope resolution operator, passing objects as arguments, returning objects, and object assignment.

[8 hrs] (CO1)

Module 2: Classes & Objects –FII

Constructors, Destructors, friend functions, Parameterized constructors, Static data members, Functions, Arrays of objects, Pointers to objects, this pointer, and reference parameter, Dynamic allocation of objects, Copyconstructors, Operator overloading using friend functions, overloading.

[8 hrs] (CO1, CO2)

Module 3: Inheritance

Base Class, Inheritance and protected members, Protected base class inheritance, Inheriting multiple base classes, Constructors, Destructors and Inheritance, Passing parameters to base class constructors, Granting access, Virtual base classes.

[8 hrs] (CO3, CO4)

Module 4: Virtual functions, Polymorphism

Virtual function, calling a Virtual function through a base class reference, Virtual attribute is inherited, Virtual functions are hierarchical, pure virtual functions, Abstract classes, Using virtual functions, Early and late binding.

[8 hrs] (CO3, CO4)

Module 5: Exception Handling

Basics of exception handling, exception handling mechanism, throwing mechanism, catching mechanism, I/O System Basics, File I/0: Exception handling fundamentals, Exception handling options. C++ stream classes, Formatted I/O, fstream and the File classes, Opening and closing a file, Reading and writing text files.

[10 hrs] (CO5)

Course Outcomes:

The student will be able to:

1. Identify classes, objects, members of a class and the relationships among them needed to solve a specific problem;
2. Demonstrate the concept of constructors and destructors. And create new definitions for some of the operators;
3. Create function templates, overload function templates;
4. Understand and demonstrate the concept of data encapsulation, inheritance, polymorphism with virtual functions; &
5. Demonstrate the concept of file operations, streams in C++ and various I/O manipulators.

Suggested Books:

1. E. Balagurusamy, Object Oriented Programming with C++, Tata McGraw Hill.

Reference Books:

1. Stanley B.Lippmann, JoseeLajoie: C++ Primer, 4th Edition, Addison Wesley, 2012.
2. Herbert Schildt: The Complete Reference C++, 4th Edition, Tata McGraw Hill, 2011.

Course Code: BTCS303-18	**Course Title:** Data Structure & AlgorithmsLab	**0L:0T:4P**	**2Credits**

List of Experiment:

Task 1: Write a program to insert a new element at end as well as at a given position in an array.

Task 2: Write a program to delete an element from a given whose value is given or whose position is given.

Task 3: Write a program to find the location of a given element using Linear Search.

Task 4: Write a program to find the location of a given element using Binary Search.

Task 5: Write a program to implement push and pop operations on a stack using linear array.

Task 6: Write a program to convert an infix expression to a postfix expression using stacks.

Task 7: Write a program to evaluate a postfix expression using stacks.

Task 8: Write a recursive function for Tower of Hanoi problem.

Task 9: Write a program to implement insertion and deletion operations in a queue using linear array.

Task 10: Write a menu driven program to perform following insertion

operations in a single linked list:

i. Insertion at beginning
ii. Insertion at end
iii. Insertion after a given node
iv. Traversing a linked list

Task 11: Write a menu driven program to perform following deletion operations in a single linked list:

i. Deletion at beginning
ii. Deletion at end
iii. Deletion after a given node

Task 12: Write a program to implement push and pop operations on a stack using linked list.

Task 13: Write a program to implement push and pop operations on a queue using linked list.

Task 14: Program to sort an array of integers in ascending order using bubble sort.

Task 15: Program to sort an array of integers in ascending order using selection sort.

Task 16: Program to sort an array of integers in ascending order using insertion sort.

Task 17: Program to sort an array of integers in ascending order using quick sort.

Task 18: Program to traverse a Binary search tree in Pre-order, In-order and Post-order.

Task 19: Program to traverse graphs using BFS.

Task 20: Program to traverse graphs using DFS.

Lab Outcomes:

The student will be able to:

1. Improve practical skills in designing and implementing basic linear data structure algorithms;
2. Improve practical skills in designing and implementing Non-linear data structure algorithms;
3. Use Linear and Non-Linear data structures to solve relevant problems;
4. Choose appropriate Data Structure as applied to specific problem definition; &
5. Implement Various searching algorithms and become familiar with their design methods.

Reference Books:

1. "Data Structures with C (Schaum's Outline Series)", Seymour Lipschutz, 1st edition,McGraw Hill Education.

Course Code: BTCS304-18	**Course Title:** Object Oriented Programming Lab	**0L:0T:4P**	**2Credits**

List of Experiment:

Task 1: Write a program that uses a class where the member functions are defined inside a class.

Task 2: Write a program that uses a class where the member functions are defined outside a class.

Task 3: Write a program to demonstrate the use of static data members.

Task 4: Write a program to demonstrate the use of const data members.

Task 5: Write a program to demonstrate the use of zero argument and parameterized constructors.

Task 6: Write a program to demonstrate the use of dynamic constructor.

Task 7: Write a program to demonstrate the use of explicit constructor.

Task 8: Write a program to demonstrate the use of initializer list.

Task 9: Write a program to demonstrate the overloading of increment and decrement operators.

Task 10: Write a program to demonstrate the overloading of memory management operators.

Task 11: Write a program to demonstrate the typecasting of basic type to class type.

Task 12: Write a program to demonstrate the typecasting of class type to basic type.

Task 13: Write a program to demonstrate the typecasting of class type to class type.

Task 14: Write a program to demonstrate the multiple inheritances.

Task 15: Write a program to demonstrate the runtime polymorphism.

Task 16: Write a program to demonstrate the exception handling.

Task 17: Write a program to demonstrate the use of class template.

Task 18: Write a program to demonstrate the reading and writing of mixed type of data.

Lab Outcomes:

The student will be able to:

1. Develop classes incorporating object-oriented techniques;
2. Design and implement object-oriented concepts of inheritance and polymorphism;
3. Illustrate and implement STL class of containers and need for exceptions to handle errors for object oriented programs; &
4. Design and implement any real world based problem involving GUI interface using object-oriented concepts.

Reference Books:

1. Stanley B.Lippmann, JoseeLajoie: C++ Primer, 4th Edition, Addison Wesley, 2012.
2. E. Balagurusamy, Object Oriented Programming with C++, Tata McGraw Hill.

BTAM304-18	**Mathematics Paper-III (Calculus and Ordinary Differential Equations)**	**4L:1T:0P**	**4 credits**

Detailed Contents:

Module 1:

Limit, continuity for functions with severable variables, partial derivatives, total derivative, Maxima, minima and saddle points; Method of Lagrange multipliers, Multiple Integration: double and triple integrals (Cartesian and polar), Change of order of integration in double integrals, Change of variables (Cartesian to polar), Applications of double and triple integrals to find surface area and volumes. [CO1, CO2] **(12Hrs)**

Module 2:

Sequence and series, Bolzano Weirstrass Theorem, Cauchy convergence criterion for sequence, uniform convergence, convergence of positive term series: comparison test, limit comparison test, D'Alembert's ratio test, Raabe's test, Cauchy root test, p-test, Cauchy integral test, logarithmic test, Alternating series, Leibnitz test, Power series, Taylor's series, Series for exponential, trigonometric and logarithmic functions. [CO3] **(13Hrs.)**

Module 3:

Exact, linear and Bernoulli's equations, Euler's equations, Equations not of first degree: equations solvable for p, equations solvable for y, equations solvable for x and Clairaut's type.

[CO4**] (12 hrs.)**

Module 4:

Second and higher order linear differential equations with constant coefficients, method of variation of parameters, Equations reducible to linear equations with constant coefficients: Cauchy and Legendre's equations. [CO5] **(12 hrs.)**

Course Outcomes: At the end of the course, the student will be able to:

1.Understand the functions of several variables that are essential in mostbranches of engineering;

2. Apply multiple integrals to deal with areas and volumes of various structures which are quite significant in real world;

3. Formulate and solveengineering problems related to convergence, infinite series, power series and Taylor series;

4. Create, select and utilize the learnt techniques of first degree ordinary differential equationsto model real world problems &;

5. Be acquainted with the knowledge required to solve higher order ordinary differential equations.

Textbooks/References:

1. G.B. Thomas and R.L. Finney, Calculus and Analytic geometry, 9th Edition, Pearson, Reprint, 2002.
2. T. Veerarajan, Engineering Mathematics for first year, Tata McGraw-Hill, New Delhi, 2008.
3. N.P. Bali and Manish Goyal, A text book of Engineering Mathematics, Laxmi Publications, Reprint, 2010.
4. Erwin Kreyszig, Advanced Engineering Mathematics, 9th Edition, John Wiley & Sons, 2006.
5. W.E. Boyce and R.C. DiPrima, Elementary Differential Equations and Boundary Value Problems, 9th Edition, Wiley India, 2009.
6. E.A. Coddington, An Introduction to Ordinary Differential Equations, Prentice Hall India, 1995.

Development of Societies
Course code: HSMC101-18

Credits: 3

COURSE TOPICS:

2.1 Unit I: Social Development (5 hours)

1. Concepts behind the origin of Family, Clan and Society
2. Different Social Systems
3. Relation between Human being and Society
4. Comparative studies on different models of Social Structures and their evolution

2.2 Unit II: Political Development (3 hours)

1. Ideas of Political Systems as learnt from History
2. Different models of Governing system and their comparative study

2.3 Unit III: Economic Development (18 hours)

1. Birth of Capitalism, Socialism, Marxism
2. Concept of development in pre-British, British and post British period- Barter, Jajmani
3. Idea of development in current context.
4. E. F. Schumacher's idea of development, Buddhist economics.

Gandhian idea of development. Swaraj and Decentralization.

3. READINGS

3.1 TEXTBOOK:
3.2 *REFERENCE BOOKS:

4. OTHER SESSIONS

4.1 *TUTORIALS:
4.2 *LABORATORY:
4.3 *PROJECT: Possible projects in this course could be

a) Interact with local communities and understand their issues.
b) Study local cottage industry and agricultural practices. Role of engineering and specialized knowledge.
c) Evaluation of technology in the context of its application. Social impact of technology. Environmental impact of technology. Evaluation from a holistic perspective.

PHILOSOPHY Course code: HSMC102-18

Credits: 3

COURSE TOPICS:

2.1 Unit 1:
The difference between knowledge (Vidya) and Ignorance (Avidya):
a. Upanishads;
b. Six systems orthodox and Heterodox Schools of Indian Philosophy.
c. Greek Philosophy:

2.2 Unit 2:
Origin of the Universe:

- NasidiyaSukta: "Who really knows?"
- Brhadaranyaka Upanishad; Chandogya Upanishad: Non-self, Self, real and unreal.
- Taittiriya Upanishad: SikshaValli.
- Plato's Symposium: Lack as the source of desire and knowledge.
- Socratic's method of knowledge as discovery.
- Language: Word as root of knowledge (Bhartrahari'sVakyapadiyam)
- Fourteen Knowledge basis as a sources of Vidya: Four Vedas; Six auxiliary sciences (Vedangas); Purana, Nyaya, Mimamsa and Dharma Sastras.

2.3 Unit 3:
Knowledge as Power: Francis Bacon. Knowledge as both power and self-realization in Bagavad Gita.

2.4 Unit 4:
Knowledge as oppression: M. Foucault. Discrimination between Rtam and Satyam in Indian Philosophy.

2.5 Unit 5:
Knowledge as invention: Modern definition of creativity; scientific activity in the claim that science invents new things at least through technology.

2.6 Unit 6:
Knowledge about the self, transcendental self; knowledge about society, polity and nature.

2.7 Unit 7:
Knowledge about moral and ethics codes.

2.8 Unit 8:
Tools of acquiring knowledge: Tantrayuktis, a system of inquiry (Caraka, Sushruta, Kautilya, Vyasa)

3. READINGS

1. Copleston, Frederick, History of Philosophy, Vol. 1.Great Britain: Continuum.
2 Hiriyanna, M. Outlines of Indian Philosophy, MotilalBanarsidass Publishers; Fifth Reprint edition (2009)
3 Sathaye, Avinash, Translation of NasadiyaSukta
4. Ralph T. H. Griffith. The Hymns of the Ṛgveda. MotilalBanarsidass: Delhi: 1973.
5. Raju, P. T. Structural Depths of Indian Thought, Albany: State University of New York Press.
6. Plato, Symposium, Hamilton Press.
7. KautilyaArtha Sastra. Penguin Books, New Delhi.
8. Bacon, Nova Orgum
9. Arnold, Edwin. The Song Celestial.
10. Foucault, Knowledge/Power.
11. Wildon, Anthony, System of Structure.
12. Lele, W.K. The Doctrine of Tantrayukti. Varanasi: Chowkamba Series.
13. Dasgupta, S. N. History of Indian Philosophy, MotilalBanasidas, Delhi.
14. Passmore, John, Hundred Years of Philosophy, Penguin.

4. OTHER SESSIONS:

4.1 Mode of Conduct

5. ASSESSMENT (indicative only):

Ask students to do term papers, for example, writing biographical details of founders, sustainers, transmitters, modifiers, rewriters; translating monographs of less known philosophers such as K. C. Bhattacharys, Daya Krishna, Gopinath Bhattacharya; comparative study of philosophical system such as MadhyasthaDarshan.

6. OUTCOME OF THE COURSE:

Students will develop strong natural familiarity with humanities along with right understanding enabling them to eliminate conflict and strife in the individual and society. Students shall be able to relate philosophy to literature, culture, society and lived experience can be considered.

Course Code:BTES301-18	Course Title: Digital Electronics	3L:0T:0P	3Credits

Detailed Contents:

Module 1:

NUMBER SYSTEMS: Binary, Octal, Decimal, Hexadecimal. Number base conversions, 1's, 2's complements, signed Binary numbers. Binary Arithmetic, Binary codes: Weighted BCD, Gray code, Excess 3 code, ASCII.
LOGIC GATES: AND, OR, NOT, NAND, NOR, Exclusive-OR and Exclusive-NOR. Implementations of Logic Functions using gates, NAND-NOR implementations.

Module 2 :
BOOLEAN ALGEBRA: Boolean postulates and laws – De-Morgan's Theorem, Principle of Duality, Boolean expression – Boolean function, Minimization of Boolean expressions – Sum of Products (SOP), Product of Sums (POS), Minterm, Maxterm, Canonical forms, Conversion between canonical forms, Karnaugh map Minimization, Don't care conditions, Quine-McCluskey method.

Module 3:
COMBINATIONAL CIRCUITS: Design procedure – Adders, Subtractors, BCD adder, Magnitude Comparator, Multiplexer/Demultiplexer, encoder/decoder, parity checker, code converters. Implementation of combinational logic using MUX, BCD to 7 segment decoder.

SEQUENTIAL CIRCUITS: Flip flops SR, JK, T, D and Master slave, Excitation table, Edge triggering, Level Triggering, Realization of one flip flop using other flip flops. Asynchronous/Ripple counters, Synchronous counters, Modulo-n counter, Ring Counters. Design of Synchronous counters: state diagram, Circuit implementation. Shift registers.

Module 4:
MEMORY DEVICES: Classification of memories, RAM organization, Write operation, Read operation, Memory cycle. ROM organization, PROM, EPROM, EEPROM, Programmable logic array, Programmable array logic, complex Programmable logic devices (CPLDS), Field Programmable Gate Array (FPGA).

A/D & D/A CONVERTORS : Analog & Digital signals. sample and hold circuit, A/D and D/A conversion techniques (Weighted type, R-2R Ladder type, Counter Type, Dual Slope type, Successive Approximation type).

COURSE OUTCOME:At the end of course the student will be able to:

1. Demonstrate the operation of simple digital gates, identify the symbols, develop the truth table for those gates; combine simple gates into more complex circuits; change binary, hexadecimal, octal numbers to their decimal equivalent an vice versa.
2. Demonstrate the operation of a flip-flop. Design counters and clear the concept of shift registers.
3. Study different types of memories and their applications.Convert digital signal into analog and vice versa.

Suggested Readings/ Books:

- Morris Mano, **Digital Design**, Prentice Hall of India Pvt. Ltd
- Donald P.Leach and Albert Paul Malvino, **Digital Principles and Applications**, 5 ed., Tata McGraw HillPublishing CompanyLimited, New Delhi, 2003.
- R.P.Jain, **Modern Digital Electronics**, 3 ed., Tata McGraw–Hill publishing company limited, New Delhi, 2003.
- Thomas L. Floyd, **Digital Fundamentals**, Pearson Education, Inc, New Delhi, 2003
- Ronald J. Tocci, Neal S. Widmer, Gregory L. Moss, **Digital System - Principles and Applications**, PearsonEducation.
- Ghosal ,**Digital Electronics**, Cengage Learning.

Course Code:BTES302-18	Course Title: Digital Electronics Lab	0L:0T:2P	1Credits

List of Experiments:

1. To verify the Truth-tables of all logic gates.
2. To realize and verify the Half & full adder circuits using logic gates.
3. To realize Half & full subtractor circuits using logic gates.
4. To realize Encoder and Decoder circuits
5. To realize Multiplexer circuits
6. To realize 4-bit binary-gray & gray-binary converters.
7. To realize comparator circuit for two binary numbers of 2-bit each.
8. To realize Full adder & full subtractor circuits using encoder.
9. To design Full adder & full subtractor circuits using multiplexer.
10. To design and verify the Truth tables of all flip-flops.
11. To design Mod-6/Mod-9 synchronous up-down counter.

Course Outcomes

At the end of this course student will demonstrate the ability to:

1. Realize combinational circuits using logic gates.
2. Realize sequential circuits using logic gates.
3. Realize various types of Flip-flops and counters

Fourth Semester

Course Code: BTES401-18 **Course Title:** Computer Organization & Architecture **3L:0T:0P 3Credits**

Pre-requisites: Digital Electronics

Detailed Contents:

Module 1: Functional blocks of a computer

CPU, memory, input-output subsystems, control unit. Instruction set architecture of a CPU – registers, instruction execution cycle, RTL interpretation of instructions, addressing modes, instruction set. Case study – instruction set of 8085 processor.

Data representation: signed number representation, fixed and floating point representations, character representation. Computer arithmetic – integer addition and subtraction, ripple carry adder, carry look-ahead adder, etc. multiplication – shift-andadd, Booth multiplier, carry save multiplier, etc. Division restoring and non-restoring techniques, floating point arithmetic.

[10 hrs] (CO1, CO2)

Module 2: Introduction to x86 architecture.

CPU control unit design: Hardwired and micro-programmed design approaches, Case study – design of a simple hypothetical CPU.

Memory system design: semiconductor memory technologies, memory organization.

Peripheral devices and their characteristics: Input-output subsystems, I/O device interface, I/O transfers – program controlled, interrupt driven and DMA, privileged and non-privileged instructions, software interrupts and exceptions. Programs and processes –role of interrupts in process state transitions, I/O device interfaces – SCII, USB.

[12 hrs] (CO2, CO4)

Module 3: Pipelining

Basic concepts of pipelining, throughput and speedup, pipelin501e hazards.

Parallel Processors: Introduction to parallelprocessors, Concurrent access to memory and cache coherency.

[10 hrs] (CO5)

Module 4: Memory Organization

Memory interleaving, concept of hierarchical memory organization, cache memory, cache size vs. block size, mapping functions, replacement algorithms, write policies.

[10 hrs] (CO3)

Course Outcomes:

The student will be able to:

1. Understand functional block diagram of microprocessor;
2. Apply instruction set for Writingassembly language programs;
3. Design a memory module and analyze its operation by interfacing with the CPU;
4. Classify hardwired and microprogrammed control units; &
5. Understand the concept of pipelining and its performance metrics.

Suggested Books:

1. "ComputerOrganization and Architecture", Moris Mano,
2. "ComputerOrganization and Design: The Hardware/Software Interface", 5th Edition by David A. Patterson and John L. Hennessy, Elsevier.
3. "Computer Organization and Embedded Systems", 6th Edition by CarlHamacher, McGraw Hill Higher Education.

Reference Books:

1. "Computer Architecture and Organization", 3rd Edition by John P. Hayes, WCB/McGraw-Hill
2. "Computer Organization and Architecture: Designing for Performance", 10th Edition by William Stallings, Pearson Education.
3. "Computer System Design and Architecture", 2nd Edition by Vincent P. Heuring and Harry F. Jordan, Pearson Education.

Course Code: BTCS402-18	**Course Title:** Operating Systems	**3L:0T:0P**	**3Credits**

Detailed Contents:

Module 1: Introduction
Concept of Operating Systems, Generations of Operating systems, Types of Operating Systems, OS Services, System Calls, Structure of an OS - Layered, Monolithic, Microkernel Operating Systems, Concept of Virtual Machine. Case study on UNIX and WINDOWS Operating System.

[6 hrs] (CO1)

Module 2: Processes
Definition, Process Relationship, Different states of a Process, Process State transitions, Process Control Block (PCB), Context switching
Thread: Definition, Various states, Benefits of threads, Types of threads, Concept of multithreads,
Process Scheduling: Foundation and Scheduling objectives, Types of Schedulers, Scheduling criteria: CPU utilization, Throughput, Turnaround Time, Waiting Time, Response Time; Scheduling algorithms: Pre-emptive and Non-pre-emptive, FCFS, SJF, RR; Multiprocessor scheduling: Real Time scheduling: RM and EDF.

[10 hrs] (CO2, CO3)

Module 3: Inter-process Communication
Critical Section, Race Conditions, Mutual Exclusion, Hardware Solution, StrictAlternation, Peterson's Solution, TheProducer\ConsumerProblem, Semaphores,EventCounters,Monitors, Message Passing, Classical IPC Problems: Reader's & Writer Problem, Dinning Philosopher Problem etc.

[8 hrs] (CO2)

Module 4: Deadlocks
Definition, Necessary and sufficient conditions for Deadlock, Deadlock Prevention, Deadlock Avoidance: Banker's algorithm, Deadlock detection and Recovery.

[8 hrs] (CO3)

Module 5: MemoryManagement
Basicconcept,LogicalandPhysical address map, Memory allocation: Contiguous Memory allocation –Fixedandvariable partition–Internaland External fragmentation and Compaction; Paging: Principle of operation – Page allocation–Hardware support for paging, Protection and sharing, Disadvantages of paging.
Virtual Memory: Basics of Virtual Memory – Hardware and control structures – Locality of

reference, Page fault, Working Set, Dirty page/Dirty bit – Demand paging, Page Replacement algorithms: Optimal, First in First Out (FIFO), Second Chance (SC), Not recently used (NRU) and Least Recently used (LRU).

[10 hrs] (CO4)

Module 6: I/O Hardware

I/O devices, Device controllers, Direct memory access Principles of I/O Software: Goals of Interrupt handlers, Device drivers, Device independent I/O software, Secondary-Storage Structure: Disk structure, Disk scheduling algorithms

File Management: Concept of File, Access methods, File types, File operation, Directory structure, File System structure, Allocationmethods (contiguous, linked, indexed), Free Space Management (bit vector, linked list, grouping), directory implementation (linear list, hash table), efficiency and performance.

Disk Management: Disk structure, Disk scheduling - FCFS, SSTF, SCAN, C-SCAN, Disk reliability, Disk formatting, Boot-block, Bad blocks.

[8 hrs] (CO5, CO6)

Course Outcomes:

The student will be able to:

1. Explain basic operating system concepts such as overall architecture, system calls, user mode and kernel mode;
2. Distinguish concepts related to processes, threads, process scheduling, race conditions and critical sections;
3. Analyze and apply CPU scheduling algorithms, deadlock detection and prevention algorithms;
4. Examine and categorize various memory management techniques like caching, paging, segmentation, virtual memory, and thrashing;
5. Design and implement file management system; &
6. Appraise high-level operating systems concepts such as file systems, disk-scheduling algorithms and various file systems.

Suggested Books:

1. Operating System Concepts Essentials, 9th Edition by AviSilberschatz, Peter Galvin, Greg Gagne, Wiley Asia Student Edition.
2. Operating Systems: Internals and Design Principles, 5th Edition, William Stallings, Prentice Hall of India.

Reference Books:

1. Operating System: A Design-oriented Approach, 1st Edition by Charles Crowley, Irwin Publishing
2. Operating Systems: A Modern Perspective, 2nd Edition by Gary J. Nutt, Addison-Wesley
3. Design of the Unix Operating Systems, 8th Edition by Maurice Bach, Prentice-Hall of India
4. Understanding the Linux Kernel, 3rd Edition, Daniel P. Bovet, Marco Cesati, O'Reilly and Associates

Course Code: BTCS403-18	**Course Title:** Design and Analysis of Algorithms	**3L:0T:0P**	**3Credits**

Pre-requisites: Data Structures

Detailed Contents:

Module 1: Introduction
Characteristics of algorithm. Analysis of algorithm: Asymptotic analysis of complexity bounds – best, average and worst-case behavior; Performance measurements of Algorithm, Time and space trade-offs, Analysis of recursive algorithms through recurrence relations: Substitution method, Recursion tree method and Masters' theorem.

[8 hrs] (CO1)

Module 2: Fundamental Algorithmic Strategies
Brute-Force, Greedy,Dynamic Programming, Branch- and-Bound and Backtracking methodologies for the design of algorithms; Illustrations of these techniques for Problem-Solving: Bin Packing, Knap Sack, TSP.

[10 hrs] (CO1, CO2)

Module 3: Graph and Tree Algorithms

Traversal algorithms: Depth First Search (DFS) and Breadth First Search (BFS); Shortest path algorithms, Transitive closure, Minimum Spanning Tree, Topological sorting, Network Flow Algorithm.

[10 hrs] (CO3)

Module 4: Tractable and Intractable Problems
Computability of Algorithms, Computability classes – P, NP, NP-complete and NP-hard. Cook's theorem, Standard NP-complete problems and Reduction techniques.

[8 hrs] (CO5)

Module 5: Advanced Topics
Approximation algorithms, Randomized algorithms, Heuristics and their characteristics.

[6 hrs] (CO1, CO4, CO5)

Course Outcomes:
The student will be able to:

1. For a given algorithms analyze worst-case running times of algorithms based on asymptotic analysis and justify the correctness of algorithms;
2. Explain when an algorithmic design situation calls for which design paradigm (greedy/ divide and conquer/backtrack etc.);
3. Explain model for a given engineering problem, using tree or graph, and writethe corresponding algorithm to solve the problems;
4. Demonstrate the ways to analyze approximation/randomized algorithms (expected running time, probability of error); &
5. Examine the necessity for NP class based problems and explain the use of heuristic techniques.

Suggested Books:

1. Introduction to Algorithms, 4TH Edition, Thomas H Cormen, Charles E Lieserson, Ronald L Rivest and Clifford Stein, MIT Press/McGraw-Hill.
2. Data Structures and Algorithms in C++, Weiss, 4th edition, Pearson.
3. Fundamentals of Computer Algorithms – E. Horowitz, Sartaj Saini, Galgota Publications.

Reference Books

1. Algorithm Design, 1st Edition, Jon Kleinberg and ÉvaTardos, Pearson.
2. Algorithm Design: Foundations, Analysis, and Internet Examples, Second Edition, Michael T Goodrich and Roberto Tamassia, Wiley.
3. Algorithms -- A Creative Approach, 3RD Edition, UdiManber, Addison-Wesley, Reading, MA.

Course Code: BTES402-18	**Course Title:** Computer Organization & ArchitectureLab	**0L:0T:2P**	**1Credits**

List of Experiment:

Task 1: Computer Anatomy- Memory, Ports, Motherboard and add-on cards.
Task 2: Dismantling and assembling PC.
Task 3: Introduction to 8085 kit.
Task 4: 2. Addition of two 8 bit numbers, sum 8 bit.
Task 5: Subtraction of two 8 bit numbers.
Task 6: Find 1's complement of 8-bit number.
Task 7: Find 2's complement of 8-bit number.
Task 8: Shift an 8-bit no. by one bit.
Task 9: Find Largest of two 8 bit numbers.
Task 10: Find Largest among an array of ten numbers (8 bit).
Task 11: Sum of series of 8 bit numbers.
Task 12: Introduction to 8086 kit.
Task 13: Addition and subtraction of two 16 bit numbers, sum 16 bit.
Task 14: Implement of Booth's algorithm for arithmetic operations.
Task 15: Find 1's and 2's complement of 16-bit number.
Task 16: Implement simple programs using I/O based interface.

Lab Outcomes:

The student will be able to:

1. Assemble personal computer;
2. Implement the various assembly language programs for basic arithmetic and logical operations; &
3. Demonstrate the functioning of microprocessor/microcontroller based systems with I/O interface.

Reference Books:

1. Fundamentals of Microprocessors and Microcontrollersby B. Ram, Dhanpat Rai Publications.

Course Code: BTCS404-18	**Course Title:** Operating Systems Lab	**0L:0T:4P**	**2Credits**

List of Experiment:

- **Task 1:** Installation Process of various operating systems.
- **Task 2:** Implementation of CPU scheduling algorithms to find turnaround time and waiting time. a) FCFS b) SJF c) Round Robin (pre-emptive) d) Priority.
- **Task 3:** Virtualization, Installation of Virtual Machine Software and installation of Operating System on Virtual Machine.
- **Task 4:** Commands for files & directories: cd, ls, cp, md, rm, mkdir, rmdir. Creating and viewing files using cat. File comparisons. Disk related commands: checking disk free spaces. Processes in linux, connecting processes with pipes, background processing, managing multiple processes. Background process: changing process priority, scheduling of processes at command, batch commands, kill, ps, who, sleep. Printing commands, grep, fgrep, find, sort, cal, banner, touch, file. File related commands ws, sat, cut, grep.
- **Task 5:** Shell Programming: Basic of shell programming, various types of shell, Shell Programming in bash, conditional & looping statement, case statements, parameter passing and arguments, shell variables, shell keywords, creating shell programs for automate system tasks, report printing.
- **Task 6:** Implementation of Bankers algorithm for the purpose of deadlock avoidance.

Lab Outcomes:

The student will be able to:

1. Understand and implement basic services and functionalities of the operating system;
2. Analyze and simulate CPU Scheduling Algorithms like FCFS, Round Robin, SJF, and Priority;
3. Implement commands for files and directories;
4. Understand and implement the concepts of shell programming;
5. Simulate file allocation and organization techniques; &
6. Understand the concepts of deadlock in operating systems and implement them in multiprogramming system.

Reference Books:

1. Operating Systems: Design and Implementation, Albert S. Woodhull and Andrew S. Tanenbaum, Pearson Education.

Course Code: BTCS405-18	**Course Title:** Design and Analysis of Algorithms Lab	**0L:0T:4P**	**2Credit**

List of Experiment:

Task 1: Code and analyze solutions to following problem with given strategies:
i. Knap Sack using greedy approach
ii. Knap Sack using dynamic approach

Task 2: Code and analyze to find an optimal solution to matrix chain multiplication using dynamic programming.

Task 3: Code and analyze to find an optimal solution to TSP using dynamic programming.

Task 4: Implementing an application of DFS such as:
i. to find the topological sort of a directed acyclic graph
ii. to find a path from source to goal in a maze.

Task 5: Implement an application of BFS such as:
i. to find connected components of an undirected graph
ii. to check whether a given graph is bipartite.

Task 6: Code and analyze to find shortest paths in a graph with positive edge weights using Dijkstra's algorithm.

Task 7: Code and analyze to find shortest paths in a graph with arbitrary edge weights using Bellman-Ford algorithm.

Task 8: Code and analyze to find shortest paths in a graph with arbitrary edge weights using Flyods' algorithm.

Task 9: Code and analyze to find the minimum spanning tree in a weighted, undirected graph using Prims' algorithm

Task 10: Code and analyze to find the minimum spanning tree in a weighted, undirected graph using Kruskals' algorithm.

Task 11: Coding any real world problem or TSP algorithm using any heuristic technique.

Lab Outcomes:

The student will be able to:

1. Improve practical skills in designing and implementing complex problems with different techniques;
2. Understand comparative performance of strategies and hence choose appropriate, to apply to specific problem definition;
3. Implement Various tree and graph based algorithms and become familiar with their design methods; &
4. Design and Implement heuristics for real world problems.

Reference Books

1. Data Structures and Algorithms in C++, Weiss, 4th edition, Pearson
2. Data Structures and Algorithms using Python and C++, David M. Reed and John Zelle, 2009 edition (available as e book), Franklin Beedle& Associates.

UNIVERSAL HUMAN VALUES 2: UNDERSTANDING HARMONY

Course code: HSMC122-18

Credits: 3

COURSE TOPICS:

The course has 28 lectures and 14 practice sessions in 5 modules:

Module 1: Course Introduction - Need, Basic Guidelines, Content and Process for Value Education

1. Purpose and motivation for the course, recapitulation from Universal Human Values-I
2. Self-Exploration–what is it? - Its content and process; 'Natural Acceptance' and ExperientialValidation- as the process for self-exploration.
3. Continuous Happiness and Prosperity- A look at basic Human Aspirations
4. Right understanding, Relationship and Physical Facility- the basic requirements for fulfilment of aspirations of every human being with their correct priority
5. Understanding Happiness and Prosperity correctly- A critical appraisal of the current scenario.
6. Method to fulfil the above human aspirations: understanding and living in harmony at various levels.

Include practice sessions to discuss natural acceptance in human being as the innate acceptance for living with responsibility (living in relationship, harmony and co-existence) rather than as arbitrariness in choice based on liking-disliking.

Module 2: Understanding Harmony in the Human Being - Harmony in Myself!

7. Understanding human being as a co-existence of the sentient 'I' and the material 'Body'
8. Understanding the needs of Self ('I') and 'Body' - happiness and physical facility
9. Understanding the Body as an instrument of 'I' (I being the doer, seer and enjoyer)
10. Understanding the characteristics and activities of 'I' and harmony in 'I'
11. Understanding the harmony of I with the Body: Sanyam and Health; correct appraisal of Physical needs, meaning of Prosperity in detail
12. Programs to ensureSanyam and Health.

Include practice sessions to discuss the role others have played in making material goods available to me. Identifying from one's own life. Differentiate between prosperity and accumulation. Discuss program for ensuring health vs dealing with disease.

Module 3: Understanding Harmony in the Family and Society- Harmony in Human-Human Relationship

13. Understanding values in human-human relationship; meaning of Justice (nine universal values in relationships) and program for its fulfilment to ensure mutual happiness; Trust and Respect as the foundational values of relationship.
14. Understanding the meaning of Trust; Difference between intention and competence
15. Understanding the meaning of Respect, Difference between respect and differentiation; the other salient values in relationship.
16. Understanding the harmony in the society (society being an extension of family): Resolution, Prosperity, fearlessness (trust) and co-existence as comprehensive Human Goals.
17. Visualizing a universal harmonious order in society- Undivided Society,

Universal Order- from family to world family.
Include practice sessions to reflect on relationships in family, hostel and institute as extended family, real life examples, teacher-student relationship, goal of education etc. Gratitude as a universal value in relationships. Discuss with scenarios. Elicit examples from students' lives.

Module 4: Understanding Harmony in the Nature and Existence - Whole existence as Coexistence

18. Understanding the harmony in the Nature
19. Interconnectedness and mutual fulfilment among the four orders of nature - recyclability and self-regulation in nature
20. Understanding Existence as Co-existence of mutually interacting units in all-pervasive space
21. Holistic perception of harmony at all levels of existence.

Include practice sessions to discuss human being as cause of imbalance in nature (film "Home" can be used), pollution, depletion of resources and role of technology etc.

Module 5: Implications of the above Holistic Understanding of Harmony on Professional Ethics

22. Natural acceptance of human values
23. Definitiveness of Ethical Human Conduct
24. Basis for Humanistic Education, Humanistic Constitution and Humanistic Universal Order
25. Competence in professional ethics: a. Ability to utilize the professional competence for augmenting universal human order b. Ability to identify the scope and characteristics of peoplefriendly and eco -friendly production systems, c. Ability to identify and develop appropriatetechnologies and management patterns for above production systems.
26. Case studies of typical holistic technologies, management models and production systems.
27. Strategy for transition from the present state to Universal Human Order: a. At the level of individual: as socially and ecologically responsible engineers, technologists and managers b. At the level of society: as mutually enriching institutions and organizations.
28. Sum up.

Include practice Exercises and Case Studies will be taken up in Practice (tutorial) Sessions eg. to discuss the conduct as an engineer or scientist etc.

3. READINGS:

3.1 Text Book

1. Human Values and Professional Ethics by R R Gaur, R Sangal, G P Bagaria, Excel Books, New Delhi, 2010.

3.2 Reference Books

1. Jeevan Vidya: EkParichaya, A. Nagaraj, Jeevan VidyaPrakashan, Amarkantak, 1999.
2. Human Values, A.N. Tripathi, New Age Intl. Publishers, New Delhi, 2004.
3. The Story of Stuff (Book).
4. The Story of My Experiments with Truth - by Mohandas Karamchand Gandhi
5. Small is Beautiful - E. F Schumacher.

6. Slow is Beautiful - Cecile Andrews
7. Economy of Permanence - J CKumarappa
8. Bharat Mein Angreji Raj -PanditSunderlal
9. Rediscovering India - by Dharampal
10. Hind Swaraj or Indian Home Rule - by Mohandas K. Gandhi
11. India Wins Freedom - Maulana Abdul Kalam Azad
12. Vivekananda - Romain Rolland (English)
13. Gandhi - Romain Rolland (English)

OUTCOME OF THE COURSE:

By the end of the course, students are expected to become more aware of themselves, and their surroundings (family, society, nature); they would become more responsible in life, and in handling problems with sustainable solutions, while keeping human relationships and human nature in mind. They would have better critical ability. They would also become sensitive to their commitment towards what they have understood (human values, human relationship and human society). It is hoped that they would be able to apply what they have learnt to their own self in different day-to-day settings in real life, at least a beginning would be made in this direction.

This is only an introductory foundational input. It would be desirable to follow it up by

a) Faculty -student or mentor-mentee programs throughout their time with the institution.

b) Higher level courses on human values in every aspect of living. E.g. as a professional.

Course Code: EVS101-18	**Course Title: Environmental Studies-**	**L:2; T:0; P:0**	**0Credits**

.Detailed Contents

Module 1 : Natural Resources :Renewable and non-renewable resources

Natural resources and associated problems.

a) Forest resources : Use and over-exploitation, deforestation, case studies. Timber extraction, mining, dams and their effects on forest and tribal people.

b) Water resources : Use and over-utilization of surface and ground water, floods, drought, conflicts over water, dams-benefits and problems.

c) Mineral resources : Use and exploitation, environmental effects of extracting and using mineral resources, case studies.

d) Food resources : World food problems, changes caused by agriculture and overgrazing, effects of modern agriculture, fertilizer-pesticide problems, water logging, salinity, case studies.

e) Energy resources : Growing energy needs, renewable and non renewable energy sources, use of alternate energy sources. Case studies.

f) Land resources : Land as a resource, land degradation, man induced landslides, soil erosion and desertification.

- Role of an individual in conservation of natural resources.
- Equitable use of resoureces for sustainable lifestyles.
-

Module 2 : Ecosystems

Concept of an ecosystem. Structure and function of an ecosystem.

Food chains, food webs and ecological pyramids. Introduction, types, characteristic features,

structure and function of following ecosystems:

a. Forest ecosystem
b. Aquatic ecosystems (ponds, streams, lakes, rivers, oceans, estuaries)

Module 3 : Biodiversity and its conservation

- Introduction – Definition : genetic, species and ecosystem diversity.
- Biodiversity at global, National and local levels.
- Inida as a mega-diversity nation
- Hot-sports of biodiversity.
- Threats to biodiversity : habitat loss, poaching of wildlife, man-wildlife conflicts.
- Endangered and endemic species of India
-

Module 4 : Social Issues and the Environment

- From Unsustainable to Sustainable development
- Resettlement and rahabilitation of people; its problems and concerns.
- Environmental ethics : Issues and possible solutions.
- Climate change, global warming, acid rain, ozone layer depletion, Nuclear accidents and holocaust. Case Studies.
- Public awareness.

***ACTIVITIES**

Nature club (bird watching, recognizing plants at institute/at home, recognizing local animals, appreciating biodiversity

Impart knowledge and inculcate the habit of taking interest and understanding biodiversity in and around the college campus. The students should be encouraged to take interest in bird watching, recognizing local plants, herbs and local animals. The students should be encouraged to appreciate the difference in the local biodiversity in their hometown, in the place of their study and other places they visit for vacation/breaks etc.

Following activities must be included.

Identify a tree fruit flower peculiar to a place or having origin from the place.

Making high resolution big photographs of small creatures (bees, spiders, ants. mosquitos etc.) especially part of body so that people can recognize (games on recognizing animals/plants).

Videography/ photography/ information collections on specialties/unique features of different types of common creatures.

Search and explore patents and rights related to animals, trees etc. Studying miracles of mechanisms of different body systems.

1*(A) Awareness Activities:*

a) Small group meetings about water management, promotion of recycle use, generation of less waste, avoiding electricity waste
b) Slogan making event
c) Poster making event
d) Cycle rally
e) Lectures from experts
f) Plantation
g) Gifting a tree to see its full growth
h) Cleanliness drive
i) Drive for segregation of waste
i) To live with some eminent environmentalist for a week or so to understand his work vi) To work in kitchen garden for mess
j) To know about the different varieties of plants
k) Shutting down the fans and ACs of the campus for an hour or so
l) Visit to a local area to document environmental assets river/forest/grassland/hill/mountain/lake/Estuary/Wetlands

m) Visit to a local polluted site-Urban/Rural/Industrial/Agricultural
n) Visit to a Wildlife sanctuary, National Park or Biosphere Reserve

Suggested Readings

1. Agarwal, K.C. 2001 Environmental Biology, Nidi Publ. Ltd. Bikaner.
2. BharuchaErach, The Biodiversity of India, Mapin Publishing Pvt. Ltd., Ahmedabad – 380 013, India, Email:mapin@icenet.net (R)
3. Brunner R.C., 1989, Hazardous Waste Incineration, McGraw Hill Inc. 480p
4. Clark R.S., Marine Pollution, Clanderson Press Oxford (TB)
5. Cunningham, W.P. Cooper, T.H. Gorhani, E & Hepworth, M.T. 2001, Environmental Encyclopedia, Jaico Publ. House, Mumabai, 1196p
6. Hawkins R.E., Encyclopedia of Indian Natural History, Bombay Natural History Society, Bombay (R)
7. Heywood, V.H &Waston, R.T. 1995. Global Biodiversity Assessment. Cambridge Univ. Press 1140p.
8. Mhaskar A.K., Matter Hazardous, Techno-Science Publication (TB)
9. Miller T.G. Jr. Environmental Science, Wadsworth Publishing Co. (TB)
10. Odum, E.P. 1971. Fundamentals of Ecology. W.B. Saunders Co. USA, 574p
11. Townsend C., Harper J, and Michael Begon, Essentials of Ecology, Blackwell Science (TB)
12. Trivedi R.K., Handbook of Environmental Laws, Rules Guidelines, Compliances and Stadards, Vol I and II, Enviro Media (R)
13. Trivedi R. K. and P.K. Goel, Introduction to air pollution, Techno-Science Publication (TB)
14. Wanger K.D., 1998 Environmental Management. W.B. Saunders Co. Philadelphia, USA 499p

Course Code: HSMC101-18	**Course Title: Development of Societies**	**3L:0T:0P**	**3Credits**

Detailed Contents:

Unit I: Social Development (5 hours)

1. Concepts behind the origin of Family, Clan and Society
2. Different Social Systems
3. Relation between Human being and Society
4. Comparative studies on different models of Social Structures and their evolution

Unit II: Political Development (3 hours)

1. Ideas of Political Systems as learnt from History
2. Different models of Governing system and their comparative study

Unit III: Economic Development (18 hours)

1. Birth of Capitalism, Socialism, Marxism
2. Concept of development in pre-British, British and post British period- Barter, Jajmani
3. Idea of development in current context.
4. E. F. Schumacher's idea of development, Buddhist economics.

Gandhian idea of development. Swaraj and Decentralization.

PROJECT: Possible projects in this course could be

a) Interact with local communities and understand their issues.
b) Study local cottage industry and agricultural practices. Role of engineering and specialized knowledge.
c) Evaluation of technology in the context of its application. Social impact of technology. Environmental impact of technology. Evaluation from a holistic perspective.

Course Code: HSMC102-18	Course Title: PHILOSOPHY	3L:0T:0P	3Credits

Detailed Contents:

Unit 1:

The difference between knowledge (Vidya) and Ignorance (Avidya):

a. Upanishads;

b. Six systems orthodox and Heterodox Schools of Indian Philosophy.

c. Greek Philosophy:

Unit 2:

Origin of the Universe:

- NasidiyaSukta: "Who really knows?"
- Brhadaranyaka Upanishad; Chandogya Upanishad: Non-self, Self, real and unreal.
- Taittiriya Upanishad: SikshaValli.
- Plato's Symposium: Lack as the source of desire and knowledge.
- Socratic's method of knowledge as discovery.
- Language: Word as root of knowledge (Bhartrahari'sVakyapadiyam)
- Fourteen Knowledge basis as a sources of Vidya: Four Vedas; Six auxiliary sciences (Vedangas); Purana, Nyaya, Mimamsa and Dharma Sastras.

Unit 3:

Knowledge as Power: Francis Bacon. Knowledge as both power and self-realization in Bagavad Gita.

Unit 4:

Knowledge as oppression: M. Foucault. Discrimination between Rtam and Satyam in Indian Philosophy.

Unit 5:

Knowledge as invention: Modern definition of creativity; scientific activity in the claim that science invents new things at least through technology.

Unit 6:

Knowledge about the self, transcendental self; knowledge about society, polity and nature.

Unit 7:

Knowledge about moral and ethics codes.

Unit 8:

Tools of acquiring knowledge: Tantrayuktis, a system of inquiry (Caraka, Sushruta, Kautilya, Vyasa)

READINGS

1. Copleston, Frederick, History of Philosophy, Vol. 1.Great Britain: Continuum.
2 Hiriyanna, M. Outlines of Indian Philosophy, MotilalBanarsidass Publishers; Fifth Reprint edition (2009)
3 Sathaye, Avinash, Translation of NasadiyaSukta
4. Ralph T. H. Griffith. The Hymns of the Ṛgveda. MotilalBanarsidass: Delhi: 1973.
5. Raju, P. T. Structural Depths of Indian Thought, Albany: State University of New York Press.
6. Plato, Symposium, Hamilton Press.
7. KautilyaArtha Sastra. Penguin Books, New Delhi.
8. Bacon, Nova Orgum
9. Arnold, Edwin. The Song Celestial.
10. Foucault, Knowledge/Power.
11. Wildon, Anthony, System of Structure.
12. Lele, W.K. The Doctrine of Tantrayukti. Varanasi: Chowkamba Series.
13. Dasgupta, S. N. History of Indian Philosophy, MotilalBanasidas, Delhi.

14. Passmore, John, Hundred Years of Philosophy, Penguin.

ASSESSMENT (indicative only):

Ask students to do term papers, for example, writing biographical details of founders, sustainers, transmitters, modifiers, rewriters; translating monographs of less known philosophers such as K. C. Bhattacharys, Daya Krishna, Gopinath Bhattacharya; comparative study of philosophical system such as MadhyasthaDarshan.

OUTCOME OF THE COURSE:

Students will develop strong natural familiarity with humanities along with right understanding enabling them to eliminate conflict and strife in the individual and society. Students shall be able to relate philosophy to literature, culture, society and lived experience can be considered.

BTCS401-18	Discrete Mathematics	3L:1T:0P	4 Credits

Detailed contents:

Module 1:

Sets, Relation and Function: Operations and Laws of Sets, Cartesian Products, Binary Relation, Partial Ordering Relation, Equivalence Relation, Image of a Set, Sum and Product of Functions, Bijective functions, Inverse and Composite Function, Size of a Set, Finite and infinite Sets, Countable and uncountable Sets, Cantor's diagonal argument and The Power Set theorem, Schroeder-Bernstein theorem.

Principles of Mathematical Induction: The Well-Ordering Principle, Recursive definition, The Division algorithm: Prime Numbers, The Greatest Common Divisor: Euclidean Algorithm, The Fundamental Theorem of Arithmetic. CO1, CO2

Module 2:

Basic counting techniques-inclusion and exclusion, pigeon-hole principle, permutation and combination. CO3

Module 3:

Propositional Logic: Syntax, Semantics, Validity and Satisfiability, Basic Connectives and Truth Tables, Logical Equivalence: The Laws of Logic, Logical Implication, Rules of Inference, The use of Quantifiers. **Proof Techniques:** Some Terminology, Proof Methods and Strategies, Forward Proof, Proof by Contradiction, Proof by Contraposition, Proof of Necessity and Sufficiency. CO3, CO4

Module 4:

Algebraic Structures and Morphism: Algebraic Structures with one Binary Operation, Semi Groups, Monoids, Groups, Congruence Relation and Quotient Structures, Free and Cyclic Monoids and Groups, Permutation Groups, Substructures, Normal Subgroups, Algebraic Structures with two Binary Operation, Rings, Integral Domain and Fields. Boolean Algebra and Boolean Ring, Identities of Boolean Algebra, Duality, Representation of Boolean Function, Disjunctive and Conjunctive Normal Form CO4

Module 5:

Graphs and Trees: Graphs and their properties, Degree, Connectivity, Path, Cycle, Sub Graph, Isomorphism, Eulerian and Hamiltonian Walks, Graph Colouring, Colouring maps and Planar Graphs, Colouring Vertices, Colouring Edges, List Colouring, Perfect Graph, definition properties and Example, rooted trees, trees and sorting, weighted trees and prefix codes, Bi-connected component and Articulation Points, Shortest distances. CO5

Suggested books:

1. Kenneth H. Rosen, Discrete Mathematics and its Applications, Tata McGraw – Hill
2. Susanna S. Epp, Discrete Mathematics with Applications, 4th edition, Wadsworth Publishing Co. Inc.
3. C L Liu and D P Mohapatra, Elements of Discrete Mathematics A Computer Oriented Approach, 3rd Edition by, Tata McGraw – Hill.

Suggested reference books:

1. J.P. Tremblay and R. Manohar, Discrete Mathematical Structure and Its Application to Computer Science", TMG Edition, TataMcgraw-Hill
2. Norman L. Biggs, Discrete Mathematics, 2nd Edition, Oxford University Press. Schaum's Outlines Series, Seymour Lipschutz, Marc Lipson,
3. Discrete Mathematics, Tata McGraw - Hill

Course Outcomes

1. To be able to express logical sentence in terms of predicates, quantifiers, and logical connectives
2. To derive the solution for a given problem using deductive logic and prove the solution based on logical inference
3. For a given a mathematical problem, classify its algebraic structure
4. To evaluate Boolean functions and simplify expressions using the properties of Boolean algebra
5. To develop the given problem as graph networks and solve with techniques of graph theory.

Fifth Semester

Course Code: BTES 501-20	Course Title: Statistical Computing Techniques using R	3L:0T:0P	3 Credits

Course Contents:

UNIT 1: CO1, CO2 [8 Hrs.]

General introduction to computing, Using R as a calculator, Numbers, words and logicals; missing values (NA), Vectors and their attributes (names, length, type), System- and user-defined objects, Accessing data (data()). Data in the system and date outside the system (read.table, scan)

UNIT 2: CO1, CO2 [10 Hrs.]

First steps in graphics, The basics of R syntax, The R workspace, Matrices and lists, Subsetting System-defined functions; the help system, Errors and warnings; coherence of the workspace Data input and output; interface with other software packages, Writing your own code; R script Good programming practice, R syntax -- further steps The parentheses and brackets; =, == and <–

Apply-type functions Compiling and applying functions Documentation, Conditional statements Loops and iterations

UNIT 3: CO1, CO2 [8 Hrs.]

Exploratory data analysis, Range, summary, mean, variance, median, sd, histogram, box plot, scatterplot

Probability distributions, Random number generation Distributions, the practice of simulation.

UNIT 4: CO1, CO2, CO3 [8 Hrs.]

Statistical functions in R, Statistical inference, contingency tables, chi-square goodness of fit, regression, generalized linear models, advanced modelling methods, the bootstrap method to compute s.e.f

UNIT 5: CO1, CO3 [8 Hrs.]

Graphics; beyond the basics Graphics and tables, Working with larger datasets, Principles of exploratory data analysis (big data analysis)

Dataframes in R, Defining your own classes and operations Models and methods in R, Customising the user's environment

Reference Books:

1. Matloff, N. (2011). The Art of R Programming: A Tour of Statistical Software Design, William
2. Philip H. Pollock (2014). An R Companion to Political Analysis, CQ Press
3. Chihara, L. and Hesterberg, T. (2011), Mathematical statistics with resampling and R, Wiley
4. Lander, J. P. (2014) R for Everyone: Advanced Analytics and Graphics, Addison-Wesley Data & Analytics Series

Course Outcomes:

At the end of the course, students will have learned:

CO1: To use a fundamental tool for computing in the practice of quantitative analytical methods (the 'paper-and-pencil' tool of the 21st century), that can work for the small jobs (like a pocket calculator) as well as for the big jobs (complex statistical data analysis).

CO2: Programming, data handling, transformations, subsetting, exploratory data analysis, probability distributions and simulations, regression and linear models, summarising data, how to handle large data sets, effective graphics.

CO3: Modern concepts of statistics based on simulations and writing a report of a quantitative analysis.

Course Code: BTES 502-20	Course Title: Statistical Computing Techniques using R lab	0L:0T:2P	1 Credits

Details of Experiments

1. Familiarization of environments in R.
2. Perform simple arithmetic's using R.
3. Perform basic R functions.
4. Use various graphical techniques in EDA.
5. Create different charts for visualization of given set of data.
6. Find the mean, median, standard deviation and quartiles of a set of observations.
7. Find the Skewness and Kurtosis of a given dataset distribution.
8. Given the scenario, implement the Bayes rule by finding the posterior probability.
9. Find the mass function of a binomial distribution with n=20,p=0.4n=20,p=0.4. Also draw the graphs of the mass function and cumulative distribution function.
10. Generate and draw the cdf and pdf of a normal distribution with mean=10 and standard deviation=3. Use values of xx from 0 to 20 in intervals of 1.
11. Construct a scatter plot to investigate the relationship between two variables.
12. Perform the Z- test for single proportion, single mean etc.
13. Calculate the regression coefficient and obtain the lines of regression for the given data.
14. Compute confidence intervals for the mean when the standard deviation is known.
15. Perform F test
16. Perform Chi-Square test.

Course Outcomes:

The Students will try to Learn:
CO1. Data manipulation, plot the graphs and charts with the help of computing features in R Programming.
CO2. The given data Interpretation with different distribution functions
CO3. The relevance and importance of the theory in solving practical problems in the real world

……

Course Code: BTCS501-18	**Course Title:** Database Management Systems	**3L:0T:0P**	**3Credits**

Detailed Contents:

Module 1: Database system architecture

Data Abstraction, Data Independence, Data Definition Language (DDL), Data Manipulation Language (DML). Data models: Entity-relationship model, network model, relational and object oriented Data models, integrity constraints, data manipulation operations.

[7hrs] (CO1,2)

Module 2: Relational query languages

Relational algebra, Tuple and domain relational calculus, SQL3, DDL and DML constructs, Open source and Commercial DBMS - MYSQL, ORACLE, DB2, SQL server. Relational database design: Domain and data dependency, Armstrong's axioms, Normal forms, Dependency preservation, Lossless design. Query processing and optimization: Evaluation of relational algebra expressions, Query equivalence, Join strategies, Query optimization algorithms.

[10hrs] (CO2,4)

Module 3:

Storage strategies, Indices, B-trees, hashing.

[3hrs] (CO3)

Module 4: Transaction processing

Concurrency control, ACID property, Serializability of scheduling, Locking and timestamp based schedulers, Multi-version and optimistic Concurrency Control schemes, Database recovery.

[6hrs] (CO3)

Module 5: Database Security

Authentication, Authorization and access control, DAC, MAC and RBAC models, Intrusion detection, SQL injection.

[8hrs] (CO 4,5)

Module 6: Advanced Topics

Object oriented and object relational databases, Logical databases, Web databases, Distributed databases.

[8hrs] (CO 5)

Course Outcomes:

At the end of study, the student shall be able to:

CO1: write relational algebra expressions for a query and optimize the Developed expressions
CO2: design the databases using ER method and normalization.
CO3: construct the SQL queries for Open source and Commercial DBMS-MYSQL, ORACLE, and DB2.
CO4: determine the transaction atomicity, consistency, isolation, and durability.
CO5: Implement the isolation property, including locking, time stamping based on concurrency control and Serializability of scheduling.

Text Books:

1. "Database System Concepts", 6th Edition by Abraham Silberschatz, Henry F. Korth, S. Sudarshan, McGraw-Hill.

Reference Books:

1. "Principles of Database and Knowledge–Base Systems", Vol1 by J. D. Ullman, Computer Science Press.
2. "Fundamentals of Database Systems", 5th Edition by R. Elmasri and S. Navathe, Pearson Education.
3. "Foundations of Databases", Reprint by Serge Abiteboul, Richard Hull, Victor Vianu, Addison-Wesley.

Course Code: BTCS505-18	**CourseTitle:** Database management System lab	**0L:0T:2P**	**1Credits**

List of Experiments:

Task 1: Introduction to SQL and installation of SQL Server / Oracle.
Task 2: Data Types, Creating Tables, Retrieval of Rows using Select Statement, Conditional Retrieval of Rows, Alter and Drop Statements.
Task 3: Working with Null Values, Matching a Pattern from a Table, Ordering the Result of a Query, Aggregate Functions, Grouping the Result of a Query, Update and Delete Statements.
Task 4: Set Operators, Nested Queries, Joins, Sequences.
Task 5: Views, Indexes, Database Security and Privileges: Grant and Revoke Commands, Commit and Rollback Commands.

Task 6: PL/SQL Architecture, Assignments and Expressions, Writing PL/SQL Code, Referencing Non-SQL parameters.
Task 7: Stored Procedures and Exception Handling.
Task 8: Triggers and Cursor Management in PL/SQL.

***Suggested Tools* –** MySQL, DB2, Oracle, SQL Server 2012, Postgre SQL, SQL lite

Course Outcomes:

CO1: This practical will enable students to retrieve data from relational databases using SQL.

CO2: students will be able to implement generation of tables using datatypes

CO3: Students will be able to design and execute the various data manipulation queries.

CO4: Students will also learn to execute triggers, cursors, stored procedures etc.

Course Code: BTCS502-18	**Course Title:** Formal Language & Automata Theory	**3L:0T:0P**	**3Credits**

Detailed Contents

Module 1: Introduction
Alphabet, languages and grammars, productions and derivation, Chomsky hierarchy of languages.

[3hrs] (CO1)

Module 2: Regular languages and finite automata:
Regular expressions and languages, deterministic finite automata (DFA) and equivalence with regular expressions, nondeterministic finite automata (NFA) and equivalence with DFA, regular grammars and equivalence with finite automata, properties of regular languages, pumping lemma for regular languages, minimization of finite automata.

[8hrs] (CO2)

Module 3: Context-free languages and pushdown automata
Context-free grammars (CFG) and languages (CFL), Chomsky and Greibach normal forms, nondeterministic pushdown automata (PDA) and equivalence with CFG, parse trees, ambiguity in CFG, pumping lemma for context-free languages, deterministic pushdown automata, closure properties of CFLs.

[8hrs] (CO3)

Module 4: Context-sensitive languages
Context-sensitive grammars (CSG) and languages, linear bounded automata and equivalence with CSG.

[5hrs] (CO4)

Module 5: Turing machines

The basic model for Turing machines (TM), Turing recognizable (recursively enumerable) and Turing-decidable (recursive) languages and their closure properties, variants of Turing machines, nondeterministic TMs and equivalence with deterministic TMs, unrestricted grammars and equivalence with Turing machines, TMs as enumerators.

[8hrs] (CO 5)

Module 6: Undecidability & Intractablity:

Church-Turing thesis, universal Turing machine, the universal and diagonalization languages, reduction between languages and Rice s theorem, undecidable problems about languages.
Intractablity: Notion of tractability/feasibility. The classes NP and co-NP, their importance. Polynomial time many-one reduction. Completeness under this reduction. Cook-Levin theorem: NP-completeness of propositional satisfiability, other variants of satisfiability. NP-complete problems from other domains: graphs (clique, vertex cover, independent sets, Hamiltonian cycle), number problem (partition), set cover

[12hrs] (CO5)

Course Outcomes: The student will be able to:

CO1: Write a formal notation for strings, languages and machines.

CO2: Design finite automata to accept a set of strings of a language.

CO3: Design context free grammars to generate strings of context free language.

CO4: Determine equivalence of languages accepted by Push Down Automata and languages generated by context free grammars

CO5: Distinguish between computability and non-computability and Decidability and undecidability.

Text Books:

1. John E. Hopcroft, Rajeev Motwani and Jeffrey D. Ullman, Introduction to Automata Theory, Languages, and Computation, Pearson Education Asia.

Reference Books:

1. Harry R. Lewis and Christos H. Papadimitriou, Elements of the Theory of Computation, Pearson Education Asia.
2. Dexter C. Kozen, Automata and Computability, Undergraduate Texts in Computer Science, Springer.
3. Michael Sipser, Introduction to the Theory of Computation, PWS Publishing.
4. John Martin, Introduction to Languages and The Theory of Computation, Tata McGraw Hill.

Course Code: BTAIML 502-20	Course Title : Artificial Intelligence	3L:0T:0P	3 Credits

Detailed Contents:

UNIT 1: Introduction: AI problems, Agents and Environments, Structure of Agents, Problem Solving Agents Basic Search Strategies: Problem Spaces, Uninformed Search (Breadth-First, Depth-First Search, Depth-first with Iterative Deepening), Heuristic Search (Hill Climbing, Generic Best-First, A*), Constraint Satisfaction (Backtracking, Local Search)

[**8hrs] (CO 1)**

UNIT 2: Advanced Search: Constructing Search Trees, Stochastic Search, A* Search Implementation, Minimax Search, Alpha-Beta Pruning Basic Knowledge Representation and Reasoning: Propositional Logic, First-Order Logic, Forward Chaining and Backward Chaining, Introduction to Probabilistic Reasoning, Bayes Theorem

[6hrs] (CO 2)

UNIT 3: Advanced Knowledge Representation and Reasoning: Knowledge Representation Issues, Nonmonotonic Reasoning, Other Knowledge Representation Schemes Reasoning Under Uncertainty: Basic probability, Acting Under Uncertainty, Bayes' Rule, Representing Knowledge in an Uncertain Domain, Bayesian Networks

[6hrs] (CO 3)

UNIT 4: Learning: What Is Learning? Rote Learning, Learning by Taking Advice, Learning in Problem Solving, Learning from Examples, Winston's Learning Program, Decision Trees.

[6hrs]
(CO 4)

UNIT 5: Expert Systems: Representing and Using Domain Knowledge, Shell, Explanation, Knowledge Acquisition.

[6hrs] (CO 5)

Course Outcomes:

At the end of the course the student should be able to:

CO 1: Understand different types of AI agents.

CO 2: Develop different types of various AI search algorithms.

CO 3: Construct simple knowledge-based systems and to apply knowledge representation.

CO 4: Convert intermediate representation in contest to understand learning.

CO 5: Apply for various techniques for Expert Systems.

Text Book:

1. Russell, S. and Norvig, P, Artificial Intelligence: A Modern Approach, Third Edition, PrenticeHall, 2010.

Reference Books:

1. Artificial Intelligence, Elaine Rich, Kevin Knight, Shivasankar B. Nair, The McGraw Hill publications, Third Edition, 2009.
2. George F. Luger, Artificial Intelligence: Structures and Strategies for Complex Problem Solving, Pearson Education, 6th ed., 2009.

Course Code: BTAIML504-20	**Course Title Artificial Intelligence Lab**	**L:0;T:0;P:2**	**1 Credits**

Detailed List of Tasks:

1.Write a programme to conduct uninformed and informed search.

2. Write a programme to conduct game search.

3. Write a programme to construct a Bayesian network from given data.

4. Write a programme to infer from the Bayesian network.

5. Write a programme to run value and policy iteration in a grid world.

6. Write a programme to do reinforcement learning in a grid world

Lab Outcomes: At the end of the course, the students are able to:

1. Explain artificial intelligence, its characteristics and its application areas.
2. Formulate real-world problems as state space problems, optimization problems or constraint satisfaction problems.
3. Select and apply appropriate algorithms and AI techniques to solve complex problems.
4. Design and develop an expert system by using appropriate tools and techniques.

……

Course Code: BTAIML501-20	**Course Title:** Programming in Python	**3L:0T:0P**	**3 Credits**	**42 Hours**

Detailed Contents:

Module 1:

Introduction to Python Programming Language: Programming Language, History and Origin of Python Language, Features of Python, Limitations, Major Applications of Python, Getting, Installing Python, Setting up Path and Environment Variables, Running Python, First Python Program, Python Interactive Help Feature, Python differences from other languages.

Python Data Types & Input/Output: Keywords, Identifiers, Python Statement, Indentation, Documentation, Variables, Multiple Assignment, Understanding Data Type, Data Type Conversion, Python Input and Output Functions, Import command.
Operators and Expressions: Operators in Python, Expressions, Precedence, Associativity of Operators, Non Associative Operators.

[8hrs] (CO1)

Module 2:
Control Structures: Decision making statements, Python loops, Python control statements (break and continue), Asserts.
Python Native Data Types: Numbers, Lists, Tuples, Sets, Dictionary, Functions & Methods of Dictionary, Strings (in detail with their methods and operations).

[10hrs] (CO1, 3)

Module 3:
Python Functions: Functions, Advantages of Functions, Built-in Functions, User defined functions, Anonymous functions, Pass by value Vs. Pass by Reference, Recursion, Scope and Lifetime of Variables.
Python Modules: Module definition, Need of modules, Creating a module, Importing module, Path Searching of a Module, Module Reloading, Standard Modules, Python Packages.

[8hrs] (CO 1, 2,3)

Module 4:
Exception Handling: Exceptions, Built-in exceptions, Exception handling, User defined exceptions in Python.
File Management in Python: Operations on files (opening, modes, attributes, encoding, closing), read() & write() methods, tell() & seek() methods, renaming & deleting files in Python, directories in Python.
Classes and Objects: The concept of OOPS in Python, Designing classes, Creating objects, Accessing attributes, Editing class attributes, Built-in class attributes, Garbage collection, Destroying objects. **[10hrs] (CO 2, 4)**

Module 5:
Generators and Iterators: Iterators, Generators, any and all functions, with statement, data compression.
Collections: namedtuple(), deque, ChainMap, Counter, OrderDict, DefaultDict, UserDict, UserList, UserString
Python Date and Time.

[6 hrs] (CO5)

Text Books:

1. Python programming: using problem solving approach, Reema Thareja, Oxford University Press.
2. Programming in Python, Pooja Sharma, BPB Publications.

Course Outcomes:

The students should be able to:

CO1: Examine Python syntax and semantics and be fluent in the use of Python flow control and functions.
CO2: Demonstrate proficiency in handling Strings, Exceptions, and File Systems.
CO3: Create, run and manipulate Python Programs using core data structures like Lists, Dictionaries.
CO4: Interpret the concepts of Object-Oriented Programming as used in Python.
CO5: Implement exemplary applications using date and time, generators, iterators, and collections in Python.

Course Code: BTAIML503-20	**Course Title:** Programming in Python Lab	**0L:0T:2P**	**1 Credits**

Prerequisites: Students should install Python.

List of Experiments:

Task 1: Write a program to demonstrate different number data types in Python.
Task 2: Write a program to perform different Arithmetic Operations on numbers in Python.
Task 3: Write a program to create, concatenate and print a string and accessing sub-string from a given string.
Task 4: Write a python script to print the current date in the following format "Sun May 29 02:26:23 IST 2017"
Task 5: Write a program to create, append, and remove lists in python.
Task 6: Write a program to demonstrate working with tuples in python.
Task 7: Write a program to demonstrate working with dictionaries in python.
Task 8: Write a python program to find largest of three numbers.
Task 9: Write a Python program to convert temperatures to and from Celsius, Fahrenheit. [Formula: c/5 = f-32/9]
Task 10: Write a Python program to construct the following pattern, using a nested for loop

```
*
* *
* * *
* * * *
* * *
* *
*
*
```

Task 11: Write a Python script that prints prime numbers less than 20.
Task 12: Write a python program to find factorial of a number using Recursion.
Task 13: Write a program that accepts the lengths of three sides of a triangle as inputs. The program output should indicate whether or not the triangle is a right triangle (Recall from the Pythagorean Theorem that in a right triangle, the square of one side equals the sum of the squares of the other two sides).
Task 14: Write a python program to define a module to find Fibonacci Numbers and import the module to another program.

Task 15: Write a python program to define a module and import a specific function in that module to another program.

Task 16: Write a script named copyfile.py. This script should prompt the user for the names of two text files. The contents of the first file should be input and written to the second file.

Task 17: Write a program that inputs a text file. The program should print all of the unique words in the file in alphabetical order.

Task 18: Write a Python class to convert an integer to a roman numeral.

Task 19: Write a Python class to implement pow(x, n)

Task 20: Write a Python class to reverse a string word by word.

ELECTIVE-I

BTAIML 505-20	**Data Visualization Using Tableau**	**3L:0T:0P**	**3 Credits**

Course Objectives:

Students should be able to describe and implement various:-

1. To understand the tableau terminologies and its fields.
2. To explain the methodologies to create a chart.
3. To gain knowledge about the different Chart types in tableau.
4. To get used with chart types and frameworks.

Detailed contents:

UNIT 1 INTRODUCTION TO DATA VISUALIZATION AND TABLEAU:

Acquiring and Visualizing Data, Applications of Data Visualization, Key Factors of Data Visualization. Introduction to Tableau Connecting to Data in Tableau, Shaping Data for Use with Tableau, Tableau Terminology, Views of data and records, Measure, Dimension, Discrete and Continuous.

(9 hrs., CO1)

UNIT 2 CREATION OF CHARTS IN TABLEAU:

Creation of bar charts in Tableau, Aggregation, Line Graphs, Independent Axes, Date Hierarchies, Marks Cards, Encoding, Level of Detail, Filters, Calculated fields, Table Calculations: - Parameters, Level of detail expressions, Dashboards and distribution.

(10 hrs, CO2)

UNIT 3 CHART TYPES:

Spreadsheet – Highlight table, Heat Map, Dual-Axis Combination Chart, Scatter Plot, Tree Map, Spark lines, Small Multiples, Bullet graphs, Stacked area, Histogram, Box and Whisker Plot, Symbol Map, Mapbox , Filled Map , Dual axis Map, Sequential Map , Polygon Maps, Gant Chart , Waterfall Chart, Dual, Axis Slope Graphs , Donut Chart , Funnel Chart, Pace chart ,Pareto Chart , Control Chart , Dynamic Dual-Axis Bump Chart ,dumbbell Chart .

(10 hrs, CO3)

UNIT 4 DATA CONNECTIVITY, TRENDS AND FORECASTING:

Data Joins, updates, exits, updating charts, Icon-Based Navigation, Filters – Analysis using Parameters , Adding alerts to dashboards, Methodology Using Custom Shape Palettes , Tableau

Data Visualization Tips, Alternative Approaches to Pie Charts ,One-Dimensional Unit Charts , Insight Framework for Data Visualization , Steps in Insight Framework – Introduction to Data Storytelling and its elements . Trends and Forecasting – Create trend lines – Model types – Create forecast.

(13 hrs, CO4,5)

Course Outcomes: At the end of the course, students will be able to:

CO1 Infer the representation of tableau and its fields.

CO2 Explore charts that are present in tableau.

CO3 Apply the various charts used for data visualization

CO4 Apply visualization tips in charts

CO5 Learn to connect the Database to tableau and forecast the predictions.

Text Books:

1 Ryan Sleeper," Practical Tableau" O'Reilly Media, Inc, First Edition, 2018
2 Learning Tableau 2020: Create effective data visualizations, build interactive visual analytics, and transform your organization, 4th Edition, 2020

Extensive Reading:

1. https://www.datacamp.com/courses/introduction–to–data–visualization–with–python
2. https://machinelearningmastery.com/data–visualization–methods–in–python
3. https://www.kaggle.com/benhamner/python–data–visualization

BTAIML 505-20	**Data Visualization Using Tableau Lab**	**0L:0T:2P**	**1 Credits**

Laboratory Work as given in the theory curriculum as guided by the instructor.

BTAIML 509-20	Java Programming	3L:0T:0P	3 Credits

Detailed contents:

UNIT 1:

The Java Environment: Installing Java, Java Program Development, Java Source File Structure, Compilation, Executions.

Basic Language Elements: Lexical Tokens, Identifiers, Keywords, Literals, Comments, Primitive Datatypes, Operators Assignments.

5 hrs., CO1, CO5

UNIT 2:

Object Oriented Programming: Class Fundamentals, Object & Object reference, Object Life time & Garbage Collection, Creating and Operating Objects, Constructor & initialization code block, Access Control, Modifiers, methods Nested, Inner Class &Anonymous Classes, Abstract Class & Interfaces Defining Methods, Argument Passing Mechanism, Method Overloading, Recursion, dealing with Static Members, Finalize() Method, Native Method. Use of "this "reference, Use of Modifiers with Classes & Methods, Design of Accessors and Mutator Methods Cloning Objects, shallow and deep cloning, Generic Class Types.

Extending Classes and Inheritance: Use and Benefits of Inheritance in OOP, Types of Inheritance in Java, Inheriting Data members and Methods, Role of Constructors in inheritance, Overriding Super Class Methods, Use of "super", Polymorphism in inheritance, Type Compatibility and Conversion Implementing interfaces.

10 hrs., CO2, CO5

UNIT 3:

Package: Organizing Classes and Interfaces in Packages, Package as Access Protection, Defining Package, CLASSPATH Setting for Packages, Making JAR Files for Library Packages Import and Static Import Naming Convention for Packages.

Exception Handling: The Idea behind Exception, Exceptions & Errors, Types of Exception, Control Flow in Exceptions, JVM reaction to Exceptions, Use of try, catch, finally, throw, throws in Exception Handling, In-built and User Defined Exceptions, Checked and Un-Checked Exceptions.

Array & String: Defining an Array, Initializing & Accessing Array, Multi –Dimensional Array, Operation on String, Mutable & Immutable String, Using Collection Bases Loop for String, Tokenizing a String, Creating Strings using StringBuffer.

Thread: Understanding Threads, Needs of Multi-Threaded Programming, Thread Life-Cycle, Thread Priorities, Synchronizing Threads, Inter Communication of Threads, Critical Factor in Thread –DeadLock,

10 hrs., CO3, CO5

UNIT 4:

GUI Programming: Designing Graphical User Interfaces in Java, Components and Containers, Basics of Components, Using Containers, Layout Managers, AWT Components, Adding a Menu to Window, Extending GUI Features Using Swing Components, Java Utilities (java.util Package) The Collection Framework : Collections of Objects , Collection Types, Sets , Sequence, Map, Understanding Hashing, Use of ArrayList & Vector.

10 hrs., CO4, CO5

UNIT 5:

Database Programming using JDBC: Introduction to JDBC, JDBC Drivers & Architecture, CURD operation Using JDBC, Connecting to non-conventional Databases.

Java Server Technologies Servlet: Web Application Basics, Architecture and challenges of Web Application, Introduction to servlet, Servlet life cycle, Developing and Deploying Servlets, Exploring Deployment , Descriptor (web.xml), Handling Request and Response.

8 hrs., CO4, CO5

Course Outcomes: At the end of the course, students will be able to:

CO1: Analyze the necessity for Object Oriented Programming paradigm over structured programming and become familiar with the fundamental concepts in OOP like encapsulation, Inheritance and Polymorphism

CO2. Design and develop java programs, analyze, and interpret object oriented data and report results.

CO3. Design an object oriented system, AWT components and multithreaded processes as per needs and specifications.

CO4: Understand the database connectivity and design web based applications on client server model

CO5. Participate and succeed in competitive examinations like GATE, Engineering services, recruitment interviews etc.

REFERENCES:

Text Books:

1. The Complete Reference Java, Herbert Schildt, ISBN: 978-0-07163177-8, Publisher: McGraw Hill, 7th Edi.

2. Thinking in Java, Bruce Eckel, ISBN: 0-13-187248-6, Publisher: Prentice Hall 4th Edition

3. The Java Programming Languages,, Ken Arnold, ISBN-13: 978- 032134980, Publisher: Sun 4th Edition,

4. Java in Nutshell,, Benjamin,ISBN: 9781449371296, Publisher: O'Reilly Media, Inc. 6th Edi.

BTAIML 510-20	**Java Programming Lab**	**0L:0T:2P**	**1 Credits**

LIST OF EXPERIMENTS:

1. Write a Java program that implements Quick sort algorithm for sorting a list of names in ascending order

2. . Write a Java program that implements Bubble sort algorithm for sorting in descending order and also shows the number of interchanges occurred for the given set of integers.

3. Write a Java program that prompts the user for an integer and then prints out all the prime numbers up to that Integer?

4. Write a Java program that checks whether a given string is a palindrome or not. Ex: MADAM is a palindrome?

5. Write a Java program that works as a simple calculator. Use a grid layout to arrange buttons for the digits and for the +, -,*, % operations. Add a text field to display the result. Handle any possible exceptions like divided by zero.

6. Write a Java program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num 2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a Number Format Exception. If Num2 were Zero, the program would throw an Arithmetic Exception. Display the exception in a message dialog box.

7. Write a Java program for the following: Create a doubly linked list of elements. Delete a given element from the above list. Display the contents of the list after deletion.

8. a) Develop an applet in Java that displays a simple message.

b) Develop an applet in Java that receives an integer in one text field, and computes its factorial Value and returns it in another text field, when the button named “Compute” is clicked.

9. Write a Java program to create an abstract class named Shape that contains two integers and an empty method named print Area (). Provide three classes named Rectangle, Triangle, and Circle such that each one of the classes extends the class Shape. Each one of the classes contains only the method print Area () that prints the area of the given shape.

10. Suppose that a table named Table.txt is stored in a text file. The first line in the file is the header, and the remaining lines correspond to rows in the table. The elements are separated by commas. Write a java program to display the table using Labels in Grid Layout.

11. Write a Java program that handles all mouse events and shows the event name at the center of the window when a mouse event is fired (Use Adapter classes).

12 Write a Java program that implements a multi-thread application that has three threads. First thread generates random integer every 1 second and if the value is even, second thread computes the square of the number and prints. If the value is odd, the third thread will print the value of cube of the number.

13. Write a Java program that correctly implements the producer – consumer problem using the concept of interthread communication.

REFERENCE BOOKS

1. Java for Programmers, P. J. Deitel and H. M. Deitel, 10th Edition Pearson education.

2. Thinking in Java, Bruce Eckel, Pearson Education.

3. Java Programming, D. S. Malik and P. S. Nair, Cengage Learning.

4. Core Java, Volume 1, 9th edition, Cay S. Horstmann and G Cornell, Pearson.

Course Outcomes: At the end of the course, students will be able to:

CO1: Use Java compiler and eclipse platform to write and execute java program.

CO2: Understand and Apply Object oriented features and Java concepts.

CO3: Apply the concept of multithreading and implement exception handling.

CO4: Access data from a Database with java program.

CO5: Develop applications using Console I/O and File I/O,GUI applications

Note:

1. Use LINUX and MySQL for the Lab Experiments. Though not mandatory, encourage the use of Eclipse platform.

2. The list suggests the minimum program set. Hence, the concerned staff is requested to add more problems to the list as needed.

Course Code: BTAIML507-20	**Course Title:** User Interface development	**3L:0T:0P**	**3 Credits**	**42 Hours**

Detailed Contents:

UNIT 1:

The User Interface: Introduction & Overview, The importance of user interface – Defining the user interface, The importance of Good design, Characteristics of graphical and web user interfaces, Principles of user interface design. (8L)

UNIT 2:

The User Interface Design process: Obstacles, Usability, Human characteristics in Design, Human Interaction speeds, Business functions-Business definition and requirement analysis, Basic business functions, Design standards. (8L)

UNIT 3:

System menus and navigation schemes : Structures of menus, Functions of menus, Contents of menus, Formatting of menus, Phrasing the menu, Selecting menu choices, Navigating menus, Kinds of graphical menus (8L)

UNIT 4:

Windows: Characteristics, Components of window, Window presentation styles, Types of window, Window management, Organizing window functions, Window operations, Web systems, Characteristics of device based controls. (8L)

UNIT 5:

Screen based controls: Operable control, Text control, Selection control, Custom control, Presentation control, Windows Tests-prototypes, kinds of tests. (8L)

Text Book:

1. Wilbert O. Galitz, "The Essential Guide to User Interface Design", John Wiley & Sons, Second Edition 2002.

Reference Books:

1. Ben Sheiderman, "Design the User Interface", Pearson Education, 1998.

2. Alan Cooper, "The Essential of User Interface Design", Wiley- Dream Tech Ltd.,2002

Course Code: BTAIML508-20	**Course Title: user interface development** Lab	**0L:0T:2P**	**1 Credits**

Design and Development of User Interfaces using HTML, CSS, JavaScript and Angular JS / Node JS technologies

Sixth Semester

Course Code: BTCS 504-18	**Course Title:** Computer Networks	**3L:0T:0P**	**3 Credits**

Detailed Contents:

Module 1: Data Communication Components

Representation of data and its flow Networks, Various Connection Topology, Protocols and Standards, OSI model, Transmission Media, LAN: Wired LAN, Wireless LANs, Connecting LAN and Virtual LAN, Techniques for Bandwidth utilization: Multiplexing - Frequency division, Time division and Wave division, Concepts on spread spectrum.

[8hrs] (CO1)

Module 2: Data Link Layer and Medium Access Sub Layer

Error Detection and Error Correction - Fundamentals, Block coding, Hamming Distance, CRC; Flow Control and Error control protocols - Stop and Wait, Go back – N ARQ, Selective Repeat ARQ, Sliding Window, Piggybacking, Random Access, Multiple access protocols -Pure ALOHA, Slotted ALOHA, CSMA/CDCDMA/CA.

[10 hrs] (CO2)

Module 3: Network Layer

Switching, Logical addressing – IPV4, IPV6; Address mapping – ARP, RARP, BOOTP and DHCP–Delivery, Forwarding and Unicast Routing protocols.

[8 hrs] (CO3)

Module 4: Transport Layer

Process to Process Communication, User Datagram Protocol (UDP), Transmission Control Protocol (TCP), SCTP Congestion Control; Quality of Service, QoS improving techniques: Leaky Bucket and Token Bucket algorithm.

[8 hrs] (CO3)

Module 5: Application Layer

Domain Name Space (DNS), DDNS, TELNET, EMAIL, File Transfer Protocol (FTP), WWW, HTTP, SNMP, Bluetooth, Firewalls, Basic concepts of Cryptography.

[8 hrs] (CO4)

Course Outcomes: The student will be able to:

CO1: Explain the functions of the different layer of the OSI Protocol;

CO2:. Describe the function of each block of wide-area networks (WANs), local area networks (LANs) and Wireless LANs (WLANs);

CO3: Develop the network programming for a given problem related TCP/IP protocol; &

CO4: Configure DNS DDNS, TELNET, EMAIL, File Transfer Protocol (FTP), WWW, HTTP, SNMP, Bluetooth, Firewalls using open source available software and tools.

Text Books:

1. Data Communication and Networking, 4th Edition, Behrouz A. Forouzan, McGraw-Hill.
2. Data and Computer Communication, 8th Edition, William Stallings, Pearson Prentice Hall India.

Reference Books:

1. Computer Networks, 8th Edition, Andrew S. Tanenbaum, Pearson New International Edition.
2. Internetworking with TCP/IP, Volume 1, 6th Edition Douglas Comer, Prentice Hall of India.

Course Code: BTCS 619-18	**Course Title:** Machine Learning	**3L:0T:0P**	**3 Credits**

Detailed Contents:

UNIT 1: **Introduction:** Well-Posed learning problems, Basic concepts, Designing a learning system, Issues in machine learning. Types of machine learning: Learning associations, Supervised learning, Unsupervised learning and Reinforcement learning.

[4hrs] (CO 1)

UNIT 2: Data Pre-processing: Need of Data Pre-processing, Data Pre-processing Methods: Data Cleaning, Data Integration, Data Transformation, Data Reduction; Feature Scaling (Normalization and Standardization), Splitting dataset into Training and Testing set.

[4hrs] (CO 2)

UNIT 3: Regression: Need and Applications of Regression, Simple Linear Regression, Multiple Linear Regression and Polynomial Regression, Evaluating Regression Models Performance (RMSE, Mean Absolute Error, Correlation, RSquare, Accuracy with acceptable error, scatter plot, *etc*.)

[6hrs] (CO 3)

UNIT 4 Classification: Need and Applications of Classification, Logistic Regression, Decision tree, Tree induction algorithm – split algorithm based on information theory, split algorithm based on Gini index; Random forest classification, Naïve Bayes algorithm; K-Nearest Neighbours (K-NN), Support Vector Machine (SVM), Evaluating Classification Models Performance (Sensitivity, Specificity, Precision, Recall, *etc*.). **Clustering**: Need and Applications of Clustering, Partitioned methods, Hierarchical methods, Density-based methods.

[12hrs] (CO 4)

UNIT 5 Association Rules Learning: Need and Application of Association Rules Learning, Basic concepts of Association Rule Mining, Naïve algorithm, Apriori algorithm. **Artificial Neural Network:** Need and Application of Artificial Neural Network, Neural network representation and working, Activation Functions.

Genetic Algorithms: Basic concepts, Gene Representation and Fitness Function, Selection, Recombination, Mutation and Elitism.

[14hrs] (CO 5)

Course Outcomes:

After undergoing this course, the students will be able to:

CO1: Analyse methods and theories in the field of machine learning

CO2: Analyse and extract features of complex datasets

CO3: Deploy techniques to comment for the Regression

CO4: Comprehend and apply different classification and clustering techniques

CO5: Understand the concept of Neural Networks and Genetic Algorithm

Suggested Readings/ Books:

Text Books:

1. Mitchell M., T., Machine Learning, McGraw Hill (1997) 1stEdition.
2. Alpaydin E., Introduction to Machine Learning, MIT Press (2014) 3rdEdition.
3. Vijayvargia Abhishek, Machine Learning with Python, BPB Publication (2018)

Reference Books:

1. Bishop M., C., Pattern Recognition and Machine Learning, Springer-Verlag (2011) 2ndEdition.
2. Michie D., Spiegelhalter J. D., Taylor C. C., Campbell, J., Machine Learning, Neural and Statistical Classification. Overseas Press (1994).

Course Code: BTCS 507-18	**Course Title:** Computer Networks Lab	**0L:0T:2P**	**1 Credits**

List of Experiments:

Task 1: To study the different types of Network cables and network topologies.
Task 2: Practically implement and test the cross-wired cable and straight through cable using clamping tool and network lab cable tester.
Task 3: Study and familiarization with various network devices.
Task 4: Familiarization with Packet Tracer Simulation tool/any other related tool.
Task 5: Study and Implementation of IP Addressing Schemes
Task 6: Creation of Simple Networking topologies using hubs and switches
Task 7: Simulation of web traffic in Packet Tracer
Task 8: Study and implementation of various router configuration commands
Task 9: Creation of Networks using routers.
Task 10: Configuring networks using the concept of subnetting
Task 11: Practical implementation of basic network command and Network configuration commands like ping, ipconfig, netstat, tracert etc. for troubleshooting network related problems.
Task 12: Configuration of networks using static and default routes.

Course Outcomes:

The students will be able to:

CO1: Know about the various networking devices, tools and also understand the implementation of network topologies;

CO2: Create various networking cables and know how to test these cables;

CO3: Create and configure networks in packet trace rtool using various network devices and topologies;

CO4: Understand IP addressing and configure networks using the subnet in;

CO5: Configure routers using various router configuration commands.

Suggested Tools **- NS2/3, Cisco packet tracer, Netsim etc.**.

Course Code: BTCS620-18	**Course Title: Machine Learning Lab**	**L:0;T:0;2 P:1**	**Credits**

Detailed List of Tasks:

1. Implement data pre-processing
2. Deploy Simple Linear Regression
3. Simulate Multiple Linear Regression
4. Implement Decision Tree
5. Deploy Random forest classification
6. Simulate Naïve Bayes algorithm
7. Implement K-Nearest Neighbors (K-NN), k-Means
8. Deploy Support Vector Machine, Apriori algorithm
9. Simulate Artificial Neural Network
10. Implement the Genetic Algorithm code

Suggested Tools Python/R/MATLAB

ELECTIVE –II

Course Code: BTCS 702-18	Course Title: Data Mining and Data Warehousing	3L: 0T: 0P	Credits: 3

Detailed Contents:

UNIT 1:

Data Warehousing Introduction: design guidelines for data warehouse implementation, Multidimensional Models; OLAP- introduction, Characteristics, Architecture, Multidimensional view Efficient processing of OLAP Queries, OLAP server Architecture ROLAP versus MOLAP Versus HOLAP and data cube, Data cube operations, data cube computation.

Data mining: What is data mining, Challenges, Data Mining Tasks, Data: Types of Data, Data Quality, Data Pre-processing, Measures of Similarity and Dissimilarity **[10hrs]**

UNIT 2:

Data mining: Introduction, association rules mining, Naive algorithm, Apriori algorithm, direct hashing and pruning (DHP), Dynamic Item set counting (DIC), Mining frequent pattern without candidate generation (FP, growth), performance evaluation of algorithms

Classification: Introduction, decision tree, tree induction algorithms – split algorithm based on information theory, split algorithm based on Gini index; naïve Bayes method; estimating predictive accuracy of classification method **[10 hrs]**

UNIT 3:

Cluster analysis: Introduction, partition methods, hierarchical methods, density based methods, dealing with large databases, cluster software

Search engines: Characteristics of Search engines, Search Engine Functionality, Search Engine Architecture, Ranking of web pages, The search engine history, Enterprise Search, Enterprise Search Engine Software. **[10 hrs]**

UNIT 4:

Web data mining: Web Terminology and Characteristics, Locality and Hierarchy in the web, Web Content Mining, Web Usage Mining, Web Structure Mining, Web mining Software. **[8 hrs]**

Suggested Readings / Books:

1. Carlo Vercellis, Business Intelligence: Data mining and Optimization for Decision Making, WILEY.

2. Han J., Kamber M. and Pei J. , b Data mining concepts and techniques, Morgan Kaufmann Publishers (2011) 3rd ed.

3. Pudi V., Krishana P.R., Data Mining, Oxford University press, (2009) 1st ed.

4. Adriaans P., Zantinge D., Data mining, Pearsoneducation press (1996), 1st ed.

5. Pooniah P. , Data Warehousing Fundamentals, Willey interscience Publication, (2001), 1st ed.

Course Code: BTAIML609-20	Course Title: Data Mining and Data Warehousing lab	0L: 0T: 2P

List of Experiments:

Task 1: Build Data Warehouse and Explore WEKA

Task 2: Perform data preprocessing tasks and demonstrate performing association rule mining on data sets

Task 3: Demonstrate performing classification on data sets

Task 4: Demonstrate performing clustering on data sets

Task 5: Demonstrate performing Regression on data sets

Task 6: Create Credit Risk Assessment Sample Programs using suitable Credit Data set

Task 7: Create Sample Programs using Hospital Management System

Task 8: Beyond the Syllabus -Simple Project on Data Preprocessing

COURSE OUTCOMES: The students will be able to:

1. Understand the various kinds of tools.
2. Demonstrate the classification, clustering and etc. in large data sets.
3. Ability to add mining algorithms as a component to the exiting tools.
4. Ability to apply mining techniques for realistic data.

Course Code: BTAIML601-20	**Course Title:** Graph Theory	3L:0T:0P	3Credits

Detailed Contents:

Module 1: Introduction

Introduction-Discovery of graphs, Definitions, Subgraphs, Isomorphic graphs, Matrix representations of graphs, Degree of a vertex, Directed walks, paths and cycles, Connectivity in digraphs, Eulerian and Hamilton digraphs, Eulerian digraphs, Hamilton digraphs, Special graphs, Complements, Larger graphs from smaller graphs, Union, Sum, Cartesian Product, Composition, Graphic sequences, Graph theoretic model of the LAN problem, Havel-Hakimi criterion, Realization of a graphic sequence.

[10 hrs] (CO1)

Module 2:

Connected graphs and shortest paths - Walks, trails, paths, cycles, Connected graphs, Distance, Cut-vertices and cut-edges, Blocks, Connectivity, Weighted graphs and shortest paths, Weighted graphs, Dijkstra's shortest path algorithm, Floyd-Warshall shortest path algorithm.

[8 hrs] (CO1, CO2)

Module 3:

Trees- Definitions and characterizations, Number of trees, Cayley's formula, Kircho-matrix-tree theorem, Minimum spanning trees, Kruskal's algorithm, Prim's algorithm, Special classes of graphs, Bipartite Graphs, Line Graphs, Chordal Graphs, Eulerian Graphs, Fleury's algorithm, Chinese Postman problem, Hamilton Graphs, Introduction, Necessary conditions and sufficient conditions.. [8 hrs] (CO3)

Module 4:

Independent sets coverings and matchings– Introduction, Independent sets and coverings: basic equations, Matchings in bipartite graphs, Hall's Theorem, K¨onig's Theorem, Perfect matchings in graphs, Greedy and approximation algorithms. [8 hrs] (CO3, CO4)

Module 5:

Vertex Colorings- Basic definitions, Cliques and chromatic number, Mycielski's theorem, Greedy coloring algorithm, Coloring of chordal graphs, Brooks theorem, Edge Colorings, Introduction and Basics, Gupta-Vizing theorem, Class-1 and Class-2 graphs, Edge-coloring of bipartite graphs, Class-2 graphs, Hajos union and Class-2 graphs, A scheduling problem and equitable edge-coloring. [10 hrs] (CO4)

Course Outcomes:

The student will be able to:

1.Know some important classes of graph theoretic problems.

2.Be able to formulate and prove central theorems about trees, matching, connectivity, colouring and planar graphs.

3.Be able to describe and apply some basic algorithms for graphs.

4.Be able to use graph theory as a modelling tool.

Suggested Books:

1. J. A. Bondy and U. S. R. Murty. Graph Theory, volume 244 of Graduate Texts in Mathematics. Springer, 1st edition, 2008.

2. J. A. Bondy and U. S. R. Murty. Graph Theory with Applications.

Reference Books:

1. Lecture Videos: http://nptel.ac.in/courses/111106050/13

2. Introduction to Graph Theory, Douglas B. West, Pearson .

3 Schaum's Outlines Graph Theory, Balakrishnan, TMH

4. Introduction to Graph Theory, Wilson Robin j, PHI

5. Graph Theory with Applications to Engineering And Computer Science, Narsing Deo, PHI

6. Graphs - An Introductory Approach, Wilson and Watkin

Course Code: BTAIML602-20	**Course Title:** Graph Theory Lab	0L:0T:2P	1Credits

List of Experiment:

Task 1: Write a program to find the number of vertices, even vertices, odd vertices and number of edges in a Graph.

Task 2: Write a program to Find Union, Intersection and ring-sum of 2 graphs.

Task 3: Write a program to Find Minimum Spanning tree Using Prim's Algorithm.

Task 4: Write a program to Find Minimum Spanning tree Using Kruskal's Algorithm.

Task 5: Write a program to find Shortest Path between 2 Vertices using Dijkstra Algorithm.

Task 6: Write a program to find Shortest Path between every pair of vertices in a graph using Floyd-Warshall's Algorithm.

Task 7: Write a program to find Shortest Path between 2 Vertices using Bellman Ford's Algorithm.

Task 8: Write a program for finding maximum Matching for bipartite graph

Task 9: Write a program for finding maximum Matching for General Path

Task 10: Write a program to find maximum flow from source node to sink node using Ford-Fulkerson Algorithm

Lab Outcomes:

The student will be able to:

1. Develop classes incorporating object-oriented techniques;
2. Design and implement object-oriented concepts of inheritance and polymorphism;
3. Illustrate and implement STL class of containers and need for exceptions to handle errors for object oriented programs; &
4. Design and implement any real world based problem involving GUI interface using object-oriented concepts.

Reference Books:

1. Lecture Videos: http://nptel.ac.in/courses/111106050/13.
2. J. A. Bondy and U. S. R. Murty. Graph Theory with Applications.

Course Code: BTDS 603-20	**Course Title: Big Data Analytics**	**L:3 T:0 P:0**	**3 Credits**

Detailed Contents:

Module I

Introduction: Big Data Overview, The rising and importance of data sciences, Big data analytics in industry verticals CO1

Hadoop Architecture: Hadoop Architecture, Hadoop ecosystem components, Hadoop Storage: HDFS, Hadoop Processing: MapReduce Framework, Hadoop Server Roles CO2

Module II

Data Analytics Lifecycle and methodology: Business Understanding, Data Understanding, Data Preparation, Modeling, Evaluation, Communicating results, Deployment, Data exploration & preprocessing CO2

Module III

Data Analytics - Theory & Methods: Measures and evaluation, Supervised learning, Linear/Logistic regression, o Decision trees, Naïve Bayes, Unsupervised learning, K-means clustering, Association rules, Unstructured Data Analytics, Technologies & tools, Text mining, Web mining CO3

Module IV

The Endgame: Opertionalizing an Analytics project, Data Visualization Techniques, Creating final deliverables

CO4

Course Outcomes

1. Describe Big Data and its importance with its applications

2. Differentiate various big data technologies like Hadoop MapReduce, Pig, Hive, Hbase and No-SQL.

3. Apply tools and techniques to analyze Big Data.

4. Design a solution for a given problem using suitable Big Data Techniques

Text Books:

1. Hadoop: The Definitive Guide by Tom White

2. Big Data Analytics: From Strategic Planning to Enterprise Integration with Tools, Techniques, NoSQL, and Graph by David Loshin

3. Machine Learning by Tom M. Mitchell

Course Code: BTDS 604-20	**Course Title: Big Data Analytics Lab**	**L:0 T:0 P:2**	**1 Credits**

List of Experiments:

1. Hands-on with Map Reduce: Hadoop, Hive, MapR
2. Hands-on with NoSQL Databases: S3, Hadoop Distributed File System(HDFS)
3. Hands-on with Statistical Packages
4. Hands-on with Visual Data Analysis tools

Lab Outcomes:

CO1: Perform data gathering of large data from a range of data sources.
CO2: Critically analyse existing Big Data datasets and implementations, taking practicality, and usefulness metrics into consideration.
CO3: Select and apply suitable statistical measures and analyses techniques for data of various structure and content and present summary statistics
CO4: Employ advanced statistical analytical skills to test assumptions, and to generate and present new information and insights from large datasets

ELECTIVE-III

Course Code: BTAIML 603-20	Course Title: Neural Networks	3L:0 T: 0P	Credits: 3

Detailed Contents:

UNIT 1 Introduction **7 hours CO1**

What is a Neural Network?, Human Brain, Models of Neuron, Neural Networks viewed as directed graphs, Feedback, Network Architectures, Knowledge representation, Artificial Intelligence and Neural Networks.

UNIT 2 Learning Processes – 1 **6 hours CO1,3**

Introduction, Error-correction learning, Memory-based learning, Hebbian learning, Competitive learning, Boltzamann learning, Credit Assignment problem, Learning with a Teacher, Learning without a Teacher, Learning tasks, Memory, Adaptation.

Learning Processes – 2, Single Layer Perceptrons **7 hours CO3**

Statistical nature of the learning process, Statistical learning theory, Approximately correct model of learning. Single Layer Perceptrons: Introduction, Adaptive filtering problem, Unconstrained optimization techniques, Linear least-squares filters, Least-mean square algorithm, Learning curves, Learning rate annealing techniques,Perceptron, Perceptron convergence theorem, Relation between the Perceptron and Bayes classifier for a Gaussian environment.

UINT3 Multilayer Perceptrons – 1 **6 hours CO2**

Introduction, Some preliminaries, Back-propagation Algorithm, Summary of back-propagation algorithm, XOR problem, Heuristics for making the back-propagation algorithm perform better, Output representation and decision rule, Computer experiment, Feature detection, Back-propagation and differentiation.

Multilayer Perceptrons – 2 **7 hours CO2**

Hessian matrix, Generalization, approximation of functions, Cross validation, Network pruning techniques, virtues and limitations of back- propagation learning, Accelerated convergence of back propagation learning, Supervised learning viewed as an optimization problem, Convolution networks.

UNIT4 Radial-Basic Function Networks – 1 **6 hours CO2**

Introduction, Cover's theorem on the separability of patterns, Interpolation problem, Supervised learning as an ill-posed Hypersurface reconstruction problem, Regularization theory, Regularization networks, Generalized radial-basis function networks, XOR problem, Estimation of the regularization parameter**.**

Radial-Basic Function Networks – 2 **6 hours CO2,4**

Approximation properties of RBF networks, Comparison of RBF networks and multilayer Perceptrons, Kernel regression and it's relation to RBF networks, Learning strategies, Computer experiment. Optimization using Hopfield networks: Traveling salesperson problem, Solving simultaneous linear equations, Allocating documents to multiprocessors**.**

Course Outcomes: At the end of the course, students will be able to-

CO1 Understand the learning and generalisation issue in neural computation.

CO2 Understand the basic ideas behind most common learning algorithms for multilayer perceptrons, radial-basis function networks, and Kohonen self-organising maps.

CO3 Implement common learning algorithms using an existing package.

CO4 Apply neural networks to classification and recognition problems.

Text Books:

1 The Essence of Neural Networks R. Callan Prentice Hall Europe, 1999

2. Neural Networks: A Comprehensive Foundation Simon Haykin Prentice Hall, 1999.

3. Neural Networks and learning Machine Haykin, Pearson, 2005, 3rd ed.

Course Code: BTAIML 604-20	**Course Title: Neural Networks lab**	**0L:0 T: 2P**	**Credits: 1**

List of experiments

1. Write a program to perform the basics matrix operations.
2. WAP to plot the Straight line.
3. WAP to plot the Sine curve.
4. How the weight & bias value effects the output of neurons.
5. How the choice of activation function effect the output of neuron experiment with the

following function purelin(n), bimary threshold(hardlim(n) haradlims(n)) ,Tansig(n) logsig(n)

6. How the weight and biased value are able to represent a decision boundary in the feature space.
7. How the Perceptron Learning rule works for Linearly Separable Problem.
8. How the Perceptron Learning rule works for Non-Linearly Separable Problem.
9. Write a program to draw a graph with multiple curve.

Experiments can be performed in MATLAB/ Python

Course Code: BTAIML605-20	**Course Title:** Recommender System	3L:0T:0P	3Credits

Detailed Contents:

Module 1: Introduction

Introduction and basic taxonomy of recommender systems (RSs). Traditional and non-personalized RSs. Overview of data mining methods for recommender systems (similarity measures, classification, Bayes classifiers, censembles of classifiers, lustering, SVMs, dimensionality reduction). Overview of convex and linear optimization principles

[3 hrs] (CO1)

Module 2: Content-based recommender systems

The long-tail principle. Domain-specific challenges in recommender systems. Content-based recommender systems. Advantages and drawbacks. Basic component's of content-based RSs. Feature selection. Item representation Methods for learning user profiles

[3 hrs] (CO1, CO2)

Module 3: Collaborative Filtering (CF)-based RSs

Mathematical optimization in CF RSs. Optimization objective. Baseline predictor through least squares. Regularization and overfitting. Temporal models. Step-by-step solution of the RS problem **,** Nearest-neighbor collaborative filtering (CF). User-based and item-based CF, comparison. Components of neighborhood methods (rating normalization, similarity weight computation, neighborhood selection). Hybrid recommender systems.

[6 hrs] (CO3, CO4)

Module 4: Context awareness and Learning principles in RSs

Context-aware recommender systems. Contextual information models for RSs. Incorporating context in Rs. Learning to rank. Active learning in RSs. Multi-armed bandits and Reinforcement learning in RSs. Dynamic RSs.

[6 hrs] (CO4)

Module 5: User behaviour understanding in RSs

Foundations of behavioral science. User choice and decisions models. Choice models in RSs. Digital nudging and user choice engineering principles. Applications and examples for recommender systems.

[3 hrs] (CO4)

Module 6: Applications of RSs for content media, social media and communities

Music and video RSs. Datasets. Group recommender systems. Social recommendations. Recommending friends: link prediction models. Similarities and differences of RSs with task assignment in mobile crowd sensing. Social network diffusion awareness in RSs.

[6 hrs] (CO5)

Course Outcomes:

The student will be able to:

1. Understand the basic concepts of recommender systems .
2. Solve mathematical optimization problems pertaing to recommender systems .
3. Carry out performance evaluation of recommender systems based on various metrics.
4. Implement machine-learning and data-mining algorithms in recommender systems data sets.
5. Design and implement a simple recommender system.

Suggested Books:

1. C.C. Aggarwal, Recommender Systems: The Textbook, Springer, 2016.

Reference Books:

1. F. Ricci, L Rokach, B. Shapira and P.B. Kantor, Recommender systems handbook, Springer 2010.
2. J. Leskovec, A. Rajaraman and J. Ullman, Mining of massive datasets, 2nd Ed., Cambridge, 2012. (Chapter 9).

Course Code: BTAIML606-20	**Course Title:** Recommender System Lab	0L:0T:4P	2Credits

List of Experiment:

Part 1. Getting ready for recommender systems

Task1:What is a recommender?

Task 2: Taxonomy of recommender systems

Task 3: Machine learning and the Netflix Prize.

Task 4: The MovieGEEKs website.

Task 5: Building a recommender system.

Task 6: User behavior and how to collect it.

Task 7: Monitoring the system.

Task 8: Ratings and how to calculate them.

Task 9: Non-personalized recommendations.

Task 10: The user (and content) who came in from the cold.

Part 2. Recommender algorithms

Task11: Finding similarities among users and among content.

Task 12: Collaborative filtering in the neighbourhood.

Task13: Evaluating and testing your recommender.

Task14: Content-based filtering.

Task15: Finding hidden genres with matrix factorization.

Task 16: Taking the best of all algorithms: Implementing hybrid recommenders.

Task17: Ranking and learning to rank.

Task 18: Future of recommender systems.

Lab Outcomes:

The student will be able to:

1. How to collect data and how to use it when you add a recommender system to your application.

2. Learn the difference between a recommendation and an advertisement, and between a personal recommendation and a non-personal one.

3. Learn how to gather data to build your own recommender system.

4. Learned about the ecosystem and infrastructure around recommender systems.

5. Learn how collaborative filtering is implemented in MovieGEEKs.

6. Evaluating the effectiveness of a recommender algorithm.

7. Splitting data sets into training data and test data.

8. Building offline experiments to evaluate recommender systems.

9. Understanding of online testing.

Reference Books:

Practical Recommend Systems by Kim Falk.

Course Code: BTAIML607-20	**Course Title:** Advance Computing and Network Technologies	3L:0T:0P	**4 Credits**

Prerequisite: Basic knowledge of Networking and its protocols.

UNIT-I

Cloud computing concepts: Introduction to virtualization techniques, Hypervisors, Types of hypervisors, Multitenancy, Application programming interfaces (API), Elasticity and scalability. Cloud service models and its architectures, Infrastructure as a service (IaaS) architecture, Platform as a service (PaaS), Software as a service (SaaS) architecture, Comparison of cloud service delivery models. Cloud deployment models and its types- Public clouds, Private clouds, Hybrid clouds, Community clouds, Migration paths for cloud, Selection criteria for cloud deployment. Mobile Cloud Computing,

Google App Engine- Azure Services Platlbrm, Amazon EC2. Amazon 53, Migrating to the Cloud-Issues and Approaches.

[10 hrs] CO1

UNIT-2

Fog Computing: Fog Computing, Characteristics, Application Scenarios, Issues and challenges. Fog Computing Architecture: Communication and Network Model, Programming Models, Fog Architecture for smart cities, healthcare and vehicles. Fog Computing Communication Technologies: Introduction , WPAN, Short-Range Technologies, LPWAN and other medium and Long-Range Technologies.

Management and Orchestration of Network Slices in 5G, Fog, Edge, and Clouds: Introduction, Background , Network Slicing in 5G , Network Slicing in Software-Defined Clouds, Network Slicing Management in Edge and Fog , Middleware for Fog and Edge Computing, Need for Fog and Edge Computing Middleware, Clusters for Lightweight Edge Clouds , IoT Integration , Security Management for Edge Cloud Architectures. [11 hrs] CO2

UNIT-3

Wireless Networking : Primer on wireless communications and networking

Physical layer: OFDM and 802.11 (WiFi) PHY, Multi-antenna systems and MIMO, Overview of 802.11n/ac PHY including beam forming MAC layer: CSMA/CA and WiFi MAC overview, Wide bandwidth channel access techniques (802.11n/ac), Energy efficiency and rate control

Multi-gigabit wireless networks : Next generation (5G) wireless technologies, Upper Gigahertz and Terahertz wireless communications, Millimeter wave networking, Directionality and beamforming, Mobility and signal blockage, IEEE 802.11ad (60 GHz WLAN) MAC and PHY overview, Visible light communication: High-speed networking using LEDs, IEEE 802.15.7

PHY and MAC overview, Sensing through visible light, Visible light indoor localization and positioning Future mobile networks : Drone networking: Multi-UAV networks, architectures and civilian applications, Communication challenges and protocols for micro UAVs, Connected and autonomous cars: Wireless technologies for Vehicle-to-Infrastructure (V2I) and Vehicle-to-Vehicle (V2V) communications, Automotive surrounding sensing with GHz and THz signals

[11 hrs] CO3

UNIT-4

Wireless Sensor Networks (WSNs): Applications of Ad Hoc and Sensor Networks - Design Challenges in Ad hoc and Sensor Networks. Wireless Networks, Issues in Ad hoc wireless networks, Routing Protocol for Ad Hoc Wireless Networks, Classifications of Routing Protocols, Table Driven Routing Protocols - Destination Sequenced Distance Vector (DSDV), On—Demand Routing protocols. Networking concept and mac protocols - Issues in Designing a MAC Protocol for Ad Hoc Wireless Networks - Design Goals of a MAC Protocol for Ad Hoc Wireless Networks, MAC Protocols for wireless sensors Networks, Low duty cycle Protocols and Wakeup concepts, S-MAC, Contention based protocols -PAMAS schedule-based protocols —LEACH, IEEE 802.15.4. MAC protocols, Energy efficient routing challenges and issues in transport layer. Challenges for Wireless Sensor Networks

Enabling Technologies for Wireless Sensor Networks, WSN application examples, Single-Node Architecture- Hardware Components, Energy Consumption of Sensor Nodes, Network Architecture- Sensor Network Scenarios, Transceiver Design Considerations.

[11 hrs] CO4

Course Outcomes:

The student will be able to:

CO1: Understand the core concepts of the cloud computing paradigm. Analyze various cloud computing service and deployment models and apply them to solve problems on the cloud.

CO2: Understand advanced architectures of Cloud and their applications in FOG and Edge computing

CO3: Understand wireless network trends and build foundations for latest wireless and mobile networks

CO4: Explains the applications of ad hoc and wireless sensor networks and apply the knowledge to identify appropriate physical and MAC layer protocols.

Suggested Readings/ Books:

1. Anthony T. Velte, Toby J. Velte and Robert Elsenpeter, “Cloud Computing: A practical Approach”, McGraw Hill, 2010.
2. C. Siva Ram Murthy, and B. S. Manoj, “Ad Hoc Wireless Networks: Architectures and Protocols ", Pearson Education, 2008.
3. Wireless Networking Complete, by Pei Zheng et al., Morgan Kaufmann.
4. Fog and Edge Computing: Principles and Paradigms (Wiley Series on Parallel and Distributed Computing) by Rajkumar Buyya and Satish Narayana Srirama.
5. Amir Vahid Dastjerdi and Rajkumar Buyya, ―Fog Computing: Helping the Internet of Things Realize its Potential, University of Melbourne

Reference Books

1. Barrie Sosinsky, “Cloud Computing Bible”, Wiley, 2011.
2. Labiod. H, “Wireless Adhoc and Sensor Networks”, Wiley, 2008.

3. Wireless Communications: Principles and Practice, by Theodore S. Rappaport, Prentice Hall.

Vijay Madisetti, Arshdeep Bahga, Internet of Things, "A Hands on Approach", University Press.

Course Code: BTAIML608-20	**Course Title:** Advance Computing and Network Technologies Lab	0L:0T:2P	**1 Credits**

List of Experiment:

Task 1: Enlist various companies in cloud business and the corresponding services provided by them and tag them under SaaS, PaaS & IaaS.

Task 2: Implementation of Para-Virtualization using VM Ware's Workstation/ Oracle's Virtual Box and Guest O.S. Learn creation, migration, cloning and managing of virtual machines.

Task 3: Setting up a private cloud using open-source tools (Eucalyptus/Open Stack etc.).

Task 4: Network Simulator installation of wireless sensor network.

Task 5: Write TCL script for transmission between mobile nodes.

Task 6: Write TCL script for sensor nodes with different parameters.

Task 7: Generate TCL script for UDP and CBR traffic in WSN nodes.

Task 8: Generate TCL script for TCP and CBR traffic in WSN nodes.

Task 9: Write a program to transfer a file from one system to another system using TCP and UDP sockets.

Task 10: Write a program to demonstrate communication between different processes using IPC.

Task 11: Write a Program to implement Routing Information Protocol (RIP) for a set of nodes

Task 12: Create a network of multiple routers and hosts to simulate RED and Drop Tail Queuing algorithm.

Task 13: Write a program to simulate Group Communication and implement Carrier sensing techniques.

Lab Outcomes:

The student will be able to:

CO1: Identify major commercial projects in the field of cloud computing.

CO2: Build and implement Wireless Sensor Network.

CO3: Design IoT applications in different domain and be able to analyse their performance.

CO4: Implement basic IoT applications on embedded platform with cloud storage.

Suggested Tools –Matlab, Cloudsim, Arduino, Raspberry Pi, Device Hive.

Suggested Readings/ Books:

1.Andrew King, "Programming the Internet of Things: An Introduction to Building Integrated,

Device-to-Cloud IoT Solutions" O'Reilly 2021.

2.Rajesh Singh Anita Gehlot, "IoT based Projects: Realization with Raspberry Pi, NodeMCU", BPB Publications, 2020.

Semester

Seventh / Eighth

Course Code: BTCS601-18	Course Title : Compiler Design	3L:0T:0P	3Credits

Detailed Contents:

UNIT 1: Unit I Introduction to Compilers:
Structure of a compiler – Lexical Analysis – Role of Lexical Analyzer – Input Buffering – Specification of Tokens – Recognition of Tokens – Lex – Finite Automata – Regular Expressions to Automata – Minimizing DFA.

[**8 hrs., CO 1]**

Unit II :Syntax Analysis:
Role of Parser – Grammars – Error Handling – Context-free grammars – Writing a grammar, Top-Down Parsing – General Strategies Recursive Descent Parser – Predictive Parser-LL(1) Parser-Shift Reduce Parser-LR Parser-LR (0) Item Construction of SLR Parsing Table - Introduction to LALR Parser – Error Handling and Recovery in Syntax Analyzer-YACC.

[**8 hrs., CO 2]**

Unit III : Intermediate Code Generation:
Syntax Directed Definitions, Evaluation Orders for Syntax Directed Definitions, Intermediate Languages: Syntax Tree, Three Address Code, Types and Declarations, Translation of Expressions, Type Checking.

[**8 hrs., CO 3]**

Unit IV: Run-Time Environment and Code Generation:

Storage Organization, Stack Allocation Space, Access to Non-local Data on the Stack, Heap Management – Issues in Code Generation – Design of a simple Code Generator.

[**6 hrs., CO 4]**

Unit V: Code Optimization:
Principal Sources of Optimization – Peep-hole optimization – DAG- Optimization of Basic Blocks-Global Data Flow Analysis – Efficient Data Flow Algorithm.

[**6 hrs., CO 5]**

Course Outcomes:

After undergoing this course, the students will be able to:
CO1: Build concepts on lexical analysis.
CO2: Understand strategies of syntax analysis.
CO3: Learn techniques of Intermediate code generation.
CO4: Undestand code design issues and design code generator.
CO5: Design and develop optimized codes.

Suggested Readings/ Books:

1. A.V. Aho, Monica, R.Sethi, J.D.Ullman, “Compilers, Principles, Techniques and

Tools", Second Edition, Pearson Education/Addison Wesley, 2009.
2. Andrew W. Appel, "Modern Compiler Implementation in Java", Second Edition, 2009.
3. J.P. Tremblay and P.G. Sorrenson, "The Theory and Practice of Compiler Writing", McGraw Hill, 1985.

Course Code: BTAIML709-20	**Course Title:** Applied Intelligence	3L:0T:0P	3Credits

Pre-requisites: AI

Detailed Contents:

Module 1: Statistical Learning Methods

Introduction to statistical learning, Statistics fundamentals: probability, random variables, description statistics and stochastic processes, Statistical inference: estimation and testing, evaluation metrics, Bayesian methods: Naive Bayes and Bayesian Networks, Markov processes and chains, Kalman estimators, Statistical modelling and decision making: regression, mixture models and classification approaches, Case study: application of statistical learning for aerospace sector problem. [8 hrs] (CO1)

Module 2: Systems Engineering

Systems challenges, The systems process, Understanding systems, Capability need and requirements, System design and architecture, System evaluation, verification and validation, The impact of organisation on Systems Engineering, People, skills and competencies in Systems Engineering. [8 hrs] (CO1, CO2)

Module 3: Intelligent Cyber Physical Systems

Cyber-physical systems: Control, sensor and actuators, Intelligent agent and multi-agent, Intelligent robotics, Embedded systems, Connected system, Countermeasures.

[8 hrs] (CO3, CO4)

Module 4: Logic and Automated Reasoning

Introduction to logical representation and reasoning, Logical Agents, Propositional Logic, First-order Logic, Inference Algorithms, Engineering domain knowledge representation, Exercises and case studies [8 hrs] (CO3, CO4)

Module 5: Deep Learning

Artificial Neural Networks (Shallow models), Backpropagation and Training, Deep learning architectures, Convolutional Neural Networks, Recurrent neural networks, Deep learning applications: object detection, identification, classification, tracking, prediction, Introduction to Reinforcement learning, Tensorflow practical sessions on Artificial, Convolutional and Recurrent Neural Networks. [10 hrs] (CO5)

Course Outcomes:

The student will be able to:

1. Explain fundamental meaning and discuss applicability of machine learning algorithms for industrial applications.

2. Test the commonly used AI algorithms and describe their applications.
3. Implement AI algorithms, estimate their performance in a simulation environment and assess their performance for a realistic case study.
4. Judge AI implementation platforms and create deep learning applications for specific problems.
5. Assess the outcomes of the statistical learning.

Suggested Books:

1. Sternberg, R., Kaufman, J., & Grigorenko, E. (2008). Applied Intelligence. Cambridge: Cambridge University Press. doi:10.1017/CBO9780511611445

Reference Books:

1.Stuart Russell and Peter Norvig, "Artificial Intelligence: A Modern Approach" , 3rd Edition, Prentice Hall
2. Elaine Rich and Kevin Knight, "Artificial Intelligence", Tata McGraw Hill
3. Trivedi, M.C., "A Classical Approach to Artifical Intelligence", Khanna Publishing House, Delhi.
4. Saroj Kaushik, "Artificial Intelligence", Cengage Learning India,
5. David Poole and Alan Mackworth, "Artificial Intelligence: Foundations for Computational Agents", Cambridge University Press 2010

ELECTIVE IV

Course Code: BTAIML 703-20	Course Title Natural Language Processing and Information Retrieval	3L:0 T: 0P	Credits: 3

Detailed Contents:

UNIT I INTRODUCTION **CO1, 6hrs.**
Natural Language Processing tasks in syntax, semantics, and pragmatics – Issues - Applications - The role of machine learning - Probability Basics –Information theory – Collocations -N-gram Language Models - Estimating parameters and smoothing - Evaluating language models.

UNIT II MORPHOLOGY AND PART OF SPEECH TAGGING **CO1, 6hrs.**
Linguistic essentials - Lexical syntax- Morphology and Finite State Transducers - Part of speech Tagging - Rule-Based Part of Speech Tagging - Markov Models - Hidden Markov Models – Transformation based Models - Maximum Entropy Models. Conditional Random Fields

UNIT III SYNTAX PARSING **CO2, 4hrs.**
Syntax Parsing - Grammar formalisms and treebanks - Parsing with Context Free Grammars - Features and Unification -Statistical parsing and probabilistic CFGs (PCFGs)-Lexicalized PCFGs

UNIT IV SEMANTIC ANALYSIS **CO2, 6 hrs.**
Representing Meaning – Semantic Analysis - Lexical semantics –Word-sense disambiguation - Supervised – Dictionary based and Unsupervised Approaches - Compositional semantics, Semantic Role Labeling and Semantic Parsing – Discourse Analysis.

UNIT V Machine Translation (MT) **CO3, 6 hrs.**
Basic issues in MT-Statistical translation-word alignment- phrase-based translation – Question Answering.

UNIT VI Information Retrieval (IR) **CO4, 12 hrs**
Information Retrieval-1: Introduction, Design Features of Information Retrieval systems, Information Retrieval Models, Classical Information Retrieval Models, Non-classical models of IR, Alternative Models of IR, Evaluation of the IR Systems . Natural Language Processing in IR, Relation Matching, Knowledge-base Approaches, Conceptual Graphs in IR, Cross-lingual Information Retrieval.

Course Outcomes: At the end of the course, students will be able to-

1. Describe the fundamental concepts and techniques of natural language processing.
2. Distinguish among the various techniques, taking into account the assumptions, strengths, and weaknesses of each.
3. Use appropriate descriptions, visualizations, and statistics to communicate the problems and their solutions.
4. Analyze large volume text data generated from a range of real-world applications like IR

Text Books:

1. Daniel Jurafsky and James H. Martin. 2009. Speech and Language Processing: An Introduction to Natural Language Processing, Speech Recognition, and Computational Linguistics. 2nd edition. Prentice-Hall.
2. Christopher D. Manning and Hinrich Schütze. 1999. Foundations of Statistical Natural Language Processing. MIT Press.

Course Code: BTAIML 704-20	Course Title Natural Language Processing and Information Retrieval Lab	0L:0 T: 2P	Credits: 1

Objectives: To describe the techniques and algorithms used in processing (text and speech) natural languages.

List of Experiments:

1. Write a program for word analysis
2. Write a program for word generation
3. Write a program for morphology study
4. Write a program for POS tagging using hidden markov model
5. Write a program for building chunker
6. Write a program for Robust and Scalable Parsing on Noisy Text in Web documents
7. Write a program on rule based machine translation

Students can be encourage to make a mini project on Speech recognition using NLP

Reference Books:

1. James A., Natural language Understanding 2e, Pearson Education, 1994
2. Bharati A., Sangal R., Chaitanya V.. Natural language processing: a Paninian perspective, PHI, 2000
3. Siddiqui T., Tiwary U. S.. Natural language processing and Information retrieval, OUP, 2008
4. Jurafsky, Dab and Martin, James, Speech and Language Processing, Second Edition, Prentice Hall, 2008.

Course Code: BTAIML 705-20	**Course Title: Network Security Applications using AI**	**3L:0 T: 0P**	**Credits: 3**

Unit1

Need of machine learning in Network Security, Overview of machine learning classifiers: Logistic Regression, Decision Tree, Random Forest, SVM, meaning of machine learning model, training, testing, validation and cross validation of machine learning models, confusion matrix and related terms: true positive, false positive, true negative, false negative, computation and meaning of performance metrics of machine learning models: Accuracy, Precision, Recall, F1-Score, ROC and AUC, underfitting and overfitting, tuning your models for better performance

Unit2

Understanding Credit Card Fraud, Recent Credit Card Fraud incidents and their financial implications, exploring datasets for credit card fraud, investigation of shallow and deep learning models for credit card fraud detection in terms of various performance metrics

Unit3

What is e-mail spamming? Incidents of e-mail-spamming, Datasets related to e-mail spamming, Natural language processing of e-mails for spam detection, empirical analysis of machine learning models in terms of Accuracy, Precision, Recall, F1-Score, AUC

Unit4

Overview of Intrusion Detection, Network Based Intrusion, Various types of Network Intrusions, Motivation behind Network based Intrusion Detection, Datasets for Network Based Intrusion Detection, statistical characteristics of mostly used datasets, data preprocessing, comparative analysis of shallow machine learning classifiers, deep learning models for intrusion detection

Reference books and URLs:

1. Machine Learning by E. Alpaydin, MIT Press, 2010
2. Dataset for machine learning at https://www.kaggle.com/datasets
3. Dataset for Network based intrusion https://www.caida.org/catalog/datasets/completed-datasets/

Course Code: BTAIML706-20	**Course Title:** Network Security Applications using AI Lab	0L:0T:2P	1Credits

List of practical:

1. Development of fine tuned model for Credit Card Fraud detection
2. Development of fine tuned model for detection of e-mail spam
3. Development of find tuned model for Network based Intrusion Detection

4. Deployment of machine learning models
5. Empirical analysis of various machine learning models

Course Code: BTAIML707-20	**Course Title:** Robotics and Intelligent systems	3L:0T:0P	3Credits

Detailed Contents:

Module 1: Introduction

Overview and Preliminaries, Introduction, History of Robotics, Mobile Robots, Position, and Orientation, Translational and Rotational Dynamics, Flying and Swimming Robots, Articulated Robots Transformation, Path Planning, and Trajectories, Time Response of Dynamic Systems, Dynamic Effects of Feedback Control.

[8 hrs] (CO1)

Module 2:

Control Systems, Sensors and Actuators, Sensor (Tactile, Range Finders, GPS, IMU, Position Encoders). Actuators. Locomotion. Manipulators.

[8 hrs] (CO2)

Module 3:

Introduction to Optimization, Numerical Optimization, Dynamic Optimal Control, Formal Logic, Algorithms, and Incompleteness.

[8 hrs] (CO3)

Module 4:

Computers, Computing, and Sets, Probability and Statistics.

[8 hrs] (CO3)

Module 5:

Machine Learning, Introduction to Neural Networks, Neural Networks, Information, Search, and Expert Systems, State Estimation, Stochastic Control, Parameter Estimation and Adaptive Control, Task Planning and Multi-Agent System.

[10 hrs] (CO4)

Course Outcomes:

The student will be able to:

1. Gain knowledge about different types of robots
2. Understand the concepts of various kinds of sensors and their utilities.
3. Recognize different modules for understanding the concepts of optimization.
4. Understand the concepts and Machine learning.

Reference Books:

Robotics

- H. Asada and J.-J. Slotine, *Robot Analysis and Control*, J. Wiley & Sons, 1986.

- C. Asfahl, *Robots and Manufacturing Automation*, J. Wiley & Sons, 1992.
- D. Auslander, J. Ridgely, and J. Ringgenberg, *Control Software for Mechanical Systems*, Prentice-Hall, 2002.
- G. Bekey, *Autonomous Robots*, MIT Press, 2005.
- M. Brady, J. Hollerbach, T. Johnson, T. Lozano-Perez, and M. Mason, *Robot Motion: Planning and Control*, MIT Press, 1984.
- H. Choset, *Principles of Robot Motion*, MIT Press, 2005.
- C. Close and D. Frederick, *Modeling and Analysis of Dynamic Systems*, Houghton Mifflin, 1993.

Intelligent Systems

- Albus, J. I., and Meystel, A. M., *Engineering of Mind*, J. Wiley & Sons, 2001.
- P. Antsaklis and K. Passino, *An Introduction to Intelligent and Autonomous Control*, Kluwer, 1993.
- R. Arkin, *Behavior-Based Robotics*, Bradford, 1998.
- P. Baldi and S. Brunak, *Bioinformatics*, Bradford, 1998.
- C. Bishop, *Neural Networks for Pattern Recognition*, Oxford University Press, 1995.
- R. Brooks, *Cambrian Intelligence*, Bradford, 1999.

Course Code: BTAIML708-20	**Course Title:** Robotics and Intelligent systems Lab	0L:0T:4P	2Credits

List of Experiments:

1. Understand the core concepts and terminologies of robotics.
2. Create 2D and 3D drawings of robots using freeware such as LibreCAD and Blender.
3. Simulate your robot using ROS and Gazebo.
4. Build robot hardware from the requirements.
5. Explore a diverse range of actuators and its interfacing.
6. Interface various robotic sensors to robots.
7. Set up and program OpenCV, OpenNI, and PCL to process 2D/3D visual data.
8. Learn speech processing and synthesis using Python.
9. Apply artificial intelligence to robots using Python.
10. Integrating Robotic Hardware and Software.
11. Build a robot control GUI using Qt and Python.
12. Calibration and testing of robot.

Reference Books:

1. Joseph, Lentin. learning Robotics using python. Packt Publishing Ltd, 2015.

ELECTIVE V

Course Code: BTCS 704-18	Course Title : Deep Learning	3L:0T:0P	3Credits

Detailed Contents:

UNIT 1: **Machine Learning Basics:** Learning, Under-fitting, Overfitting, Estimators, Bias, Variance, Maximum Likelihood Estimation, Bayesian Statistics, Supervised Learning, Unsupervised Learning and Stochastic Gradient Decent.

[4hrs] (CO 1)

UNIT 2: Deep Feedforward Network: Feed-forward Networks, Gradient-based Learning, Hidden Units, Architecture Design, Computational Graphs, Back-Propagation, Regularization, Parameter Penalties, Data Augmentation, Multi-task Learning, Bagging, Dropout and Adversarial Training and Optimization.

[4hrs] (CO 2)

UNIT 3: Convolution Networks: Convolution Operation, Pooling, Basic Convolution Function, Convolution Algorithm, Unsupervised Features and Neuroscientific for convolution Network.

[6hrs] (CO 3)

UNIT 4: Sequence Modelling: Recurrent Neural Networks (RNNs), Bidirectional RNNs, Encoder- Decoder Sequence-to-Sequence Architectures, Deep Recurrent Network, Recursive Neural Networks and Echo State networks.

[12hrs] (CO 4)

UNIT 5: Deep Generative Models: Boltzmann Machines, Restricted Boltzmann Machines, Deep Belief Networks, Deep Boltzmann Machines, Sigmoid Belief Networks, Directed Generative Net, Drawing Samples from Auto – encoders.

[14hrs] (CO 5)

Course Outcomes:

After undergoing this course, the students will be able to:
CO1: Comprehend the advancements in learning techniques
CO2: Compare and explain various deep learning architectures and algorithms.
CO3: Demonstrate the applications of Convolution Networks
CO4: Apply Recurrent Network for Sequence Modelling
CO5: Deploy the Deep Generative Models

Suggested Readings/ Books:

Text Books:

1. *Goodfellow L., Bengio Y. and Courville A., Deep Learning, MIT Press (2016).*

2. Patterson J. and Gibson A., Deep Learning: A Practitioner's Approach, O'Reilly (2017), 1st ed.

Reference Books:

1. Haykin S., Neural Network and Machine Learning, Prentice Hall Pearson (2009), 3rd ed.
2. Geron A., Hands-on Machine Learning with Sci-kit and TensorFlow, O'Reilly Media (2017)

……..……

Course Code: BTCS 705-18	**Course Title: Deep Learning Lab**	**L:0;T:0; 2P:**	**Credits;1**

Detailed List of Tasks:

- Creating a basic network and analyze its performance
- Deploy the Confusion matrix and simulate for Overfitting
- Visualizing a neural network
- Demo: Object Detection with pre-trained RetinaNet with Keras
- Neural Recommender Systems with Explicit Feedback
- Backpropagation in Neural Networks using Numpy
- Neural Recommender Systems with Implicit Feedback and the Triplet Loss
- Fully Convolutional Neural Networks
- ConvNets for Classification and Localization
- Text Classification and Word Vectors
- Character Level Language Model (GPU required)

Suggested Tools Python/R/MATLAB

……

Course Code: BTAIML711-20	**Course Title:** Augmented and Virtual reality	3L:0T:0P	3Credits

Course Details:

UNIT 1:
Introduction to Virtual and Augmented Reality (VR/AR) Systems, Definition and Overview of VR/AR, History and Evolution of VR/AR, Applications of VR/AR (4hrs.)
Technical Aspects of Virtual Reality, Virtual Reality Hardware and Peripherals, Virtual Reality Software and Development Tools, Integration of Virtual Reality Systems with Other Technologies (7hrs.)
CO1

UNIT 2:
Design and Development of VR/AR Applications, Conceptualization and Prototyping of Virtual Reality Applications, User Experience Design in Virtual Reality, Implementation of Virtual Reality Applications using Unity and Unreal Engine (15hrs.)
CO2

UNIT 3:
Evaluation of Virtual Reality Systems and Technologies, Usability and User Testing of Virtual Reality Systems, Performance Analysis and Optimization of Virtual Reality Applications, Comparison and Evaluation of Different Virtual Reality Technologies (5hrs.)
CO3

UNIT 4:

Applications of VR/AR in Various Fields, VR/AR in Education and Training, VR/AR in Entertainment and Gaming, VR/AR in Engineering and Manufacturing (7hrs.)
Challenges and Limitations in VR/AR, Emerging Trends and Future Directions in VR/AR (2hrs.)
CO4

COURSE OUTCOMES:
CO1 Learn the fundamentals concepts of virtual reality and its applications, technical concepts of virtual reality systems, including hardware and software components.
CO2 Design and develop virtual reality applications using industry-standard tools and techniques.
CO3 Evaluate the limitations and strengths of various virtual reality systems and technologies.
CO4 Apply virtual reality technology to solve real-world problems in various fields, including education, entertainment, and engineering.

Recommended Books:

1. Creating Augmented & Virtual Realities, Erin Pangilinan, Steve Lukas, Vasanth Mohan, Shroff/O'Reilly
2. Complete Virtual Reality and Augmented Reality Development with Unity: Leverage the power of Unity and become a pro at creating mixed reality applications, Jesse Glover, Jonathan Linowes, Packt Publishing Limited
3. Developing Virtual Reality Applications: Foundations of Effective Design, Alan B. Craig, Jeffrey D. Will, William R. Sherman, Morgan Kaufmann Publisher
4. Augmented Reality with Unity AR Foundation: A practical guide to cross-platform AR development with Unity 2020 and later versions, Jonathan Linowes, Packt Publishing Limited
5. 3D Game Design with Unreal Engine 4 and Blender: Design and create immersive, beautiful game environments with the versatility of Unreal Engine 4 and Blender, Justin Plowman, Ingram

Course Code: BTAIML712-20	**Course Title:** Augmented and Virtual Reality lab	L:0T:2P	1Credits

1. Installation of Unity and Visual Studio, setting up Unity for VR development, understanding documentation of the same.
2. Demonstration of the working of HTC Vive, Google Cardboard, Google Daydream and Samsung gear VR.
3. Develop a scene in Unity that includes: i. a cube, plane and sphere, apply transformations on the 3 game objects. ii. add a video and audio source.
4. Develop a scene in Unity that includes a cube, plane and sphere. Create a new material and texture separately for three Game objects. Change the colour, material and texture of each Game object separately in the scene. Write a C# program in visual studio to change the colour and material/texture of the game objects dynamically on button click.
5. Develop a scene in Unity that includes a sphere and plane . Apply Rigid body component, material and Box collider to the game Objects. Write a C# program to grab and throw the sphere using vr controller.
6. Develop a simple UI(User interface) menu with images, canvas, sprites and button. Write a C# program to interact with UI menu through VR trigger button such that on each successful trigger interaction display a score on scene ..
7. Create an immersive environment (living room/ battlefield/ tennis court) with only static game objects. 3D game objects can be created using Blender or use available 3D models.
8. Include animation and interaction in the immersive environment created in Assignment 7.

Programming tools recommended: - Unity, C#, Blender, VRTK. VR Devices: HTC Vive, Google Cardboard, Google Daydream and Samsung gear VR

Course Code: BTAIML701-20	**Course Title: Computer Vision**	**L:3;T:0; 2P**	**3Credits**

Total No. of Lectures – 42

Number of Lectures

UNIT 1	**Introduction to Digital Image Processing:** The Origins of Digital Image Processing, Elements of Visual Perception, Image Sensing and Acquisition, Image Sampling and Quantization, Relationships between pixels. **Intensity transformation and Spatial Filtering:** Image negation, Log transformations, Power law transformation, Smoothing Spatial Filters, Sharpening Spatial Filters.	**6 CO1,2**
UNIT 2	**Filtering in Frequency Domain**: Preliminary Concepts, The Discrete Fourier Transform (DFT) of One Variable, The Discrete Fourier Transform of Two Variables, Properties of 2-D DFT, Smoothing Spatial Filters, Sharpening Spatial Filters, Fast Fourier Transform. **Image Restoration:** Model for Image Degradation/Restoration Process, Noise Models, Restoration by Spatial Filtering, Restoration by Frequency Domain Filtering, Homographies	**10 CO2**
UNIT 3	**Color Image Processing:** Color Models, Pseudo color Image Processing, Color Transformations, Color Image Smoothing and Shapening. **Wavelets and Multiresolution Processing:** Image Pyramids, Haar Transform, Multiresolution Expansion, Wavelet Transform in 1-D. **Morphological Image Processing:** Erosion and Dilation, Opening and Closing, Hit or Miss Transformation, Basic Morphological Algorithms.	**12 CO2,3**
UNIT 4	**Stereo and multi-view reconstruction, Structure-from-Motion** projection matrices, camera calibration, epipolar geometry, fundamental and essential matrices, disparity maps, optical flows, volumetric shape reconstruction from window-based towards regularization-based stereo, loss functions	**10 CO4**
UNIT 5	**Object Recognition:** Pattern and Pattern Classes, Object recognition methods.	**4 CO4**

Course Outcomes: At the end of the course, students will be able to:

1. Design and implement spatial domain filters.
2. Implement smoothing and sharpening operators.
3. High pass and low pass filters for smoothing and sharpening of images.
4. Learn multi view and motion structure

Suggested Books:

Sr. No.	Name of Book/ Authors/ Publisher	Year of Publication/ Reprint
1	Digital Image Processing /Gonzalez Rafael C. and Woods Richard E./ Prentice–Hall of India	2011
2	Digital Image Processing/ Pratt William K./ PIKS Inside(3rd ed.), New Jersy: John Wiley & Sons, Inc.	2001
3	Digital Image Processing/ Bernd Jahne/ Springer	2002
4	Fundamentals of Digital Image Processing/ Annadurai S. and Shanmuga lakshmi R./ New Delhi: Pearson Education	2007
5	Digital Image Processing: An Algorithmic Approach,/ Joshi M.A./ New Delhi: Prentice Hall of India	2006

Course Code: BTAIML713-20	Course Title: Computer Vision	L:0;T:0; 2P	1Credits

Note: The implementation can be performed in MATLAB/C/C++/Java/Python on binary/grayscale/color images.

1. Implementation of basic image transformations:
    a. Log
    b. Power law
    c. Negation
2. Implementation the following:
    a. Histogram processing
    b. Histogram equalization/matching
3. Implementation of piecewise linear transformations
    a. Contrast stretching
    b. Grey level slicing
    c. Bit plane slicing
4. Implementation of image enhancement/smoothing using
    a. Linear (weighted and non weighted filters)
    b. Order statistics filters (Non linear filters)
        i. Mean
        ii. Median
        iii. Min
        iv. Max
        v. Average
5. Implementation of image enhancement/sharpening using
    a. Laplacian operators
    b. Sobel's operators
    c. Robert's cross operators
6. Implement the 2D-DFT to obtain Fourier coefficients and reconstruct the image, i.e., IDFT.
7. Image enhancement using Fourier low pass filters
    a. Ideal
    b. Butterworth
    c. Gaussian
8. Image enhancement using Fourier high pass filters
    a. Ideal
    b. Butterworth
    c. Gaussian
9. Implement algorithms to detect the following in an image
    a. Point
    b. Line
    c. Boundary
10. Implement 2D wavelet transform and IDFT.