Scheme & Syllabus of

Bachelor of Technology

Computer Science & Engg.

(Cyber Security)

Batch 2021 onwards
(3rd -8th Semester)

By

Department of Academics

IK Gujral Punjab Technical University

Bachelor of Technology in Computer Science and Engg.(Cyber security)

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 505-20	Engineering Science	Mathematics foundations for Cryptography	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
BTCS602 -18	Professional Core Courses	Artificial Intelligence	3	0	0	40	60	100	3
BTCS 504-18	Professional Core Courses	Computer Networks	3	0	0	40	60	100	3
BTITCS XXX-18	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
BTCS 505-18	Professional Core Courses	Database Management Systems Lab	0	0	4	30	20	50	2
BTCS 605-20	Professional Core Courses	Artificial Intelligence Lab	0	0	2	30	20	50	1
BTCS 507-18	Professional Core Courses	Computer Networks Lab	0	0	2	30	20	50	1
BTCS XXX-18	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**

Sixth Semester

Course Code	Type of Course	Course Title	Hours per Week			Marks Distribution		Total Marks	Credits
			L	T	P	Internal	External		
BTCS 701-18	Professional Core Courses	Network Security & Cryptography	3	0	0	40	60	100	3
BTIT CS601 -20	Professional Core Courses	Digital Forensics	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
BTIT CS602 -20	Professional Core Courses	Network Security & Cryptography Lab	0	0	2	30	20	50	1
BTIT CS603 -20	Professional Core Courses	Digital Forensics Lab	0	0	2	30	20	50	1
BTCS UUU-18	Professional Elective Courses	Elective-II lab	0	0	2	30	20	50	1
BTCS 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
BTIT CS701 -20	Professional Core Courses	Blockchain & Cryptocurrency	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 604-18	Professional Core Courses	Compiler Design Lab	0	0	2	30	20	50	1
BTIT CS701 -20	Professional Core Courses	Blockchain & Cryptocurrency Lab	0	0	2	30	20	50	1
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**	**20**	**440**	**460**	**900**	**25**

Seventh/Eighth Semester

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

*Students may be encouraged to acquire some professional certification during this like EC Council of India certified Hacker/ PT etc.

LIST OF ELECTIVES

Elective I

BTITCS 503-20 Web technologies
BTITCS 504-20 Web Technologies lab
BTAIML 501-20 Programming in Python
BTAIML 503-20 Programming in Python lab
BTITCS603-20 Cyber Law & IPR
BTITCS604-20 Cyber Law & IPR Lab

Elective II

BTITCS 501 -20 Internet of Things
BTITCS502-20 Internet of Things Lab
BTITCS703-20 Ethical Hacking
BTITCS704-20 Ethical Hacking Lab
BTITCS707-20 Cyber Attacks
BTITCS708-20 Cyber Attacks Lab

Elective III

BTCS618-18 Machine Learning
BTCS619-18 Machine Learning Lab
BTITCS605-20 Intrusion Detection System (IDS)
BTITCS606-20 Intrusion Detection System (IDS) Lab
BTITCS705-20 Vulnerability Analysis and Penetration Testing
BTITCS706-20 Vulnerability Analysis and Penetration Testing Lab

Third Semester

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 –II

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, pipeline 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.
4. For a given a mathematical problem, classify its algebraic structure
5. To evaluate Boolean functions and simplify expressions using the properties of Boolean algebra
6. To develop the given problem as graph networks and solve with techniques of graph theory.

Fifth Semester

Course Code: BTES 505-20	**Course Title: Mathematical foundations for Crptography**	**3L:0T:0P**	**3 Credits**

Detailed Contents:

Unit1:
Introduction to Number Theory :
Divisibility, TheDivisionAlgorithm, TheEuclideanAlgorithm, GreatestCommonDivisor, FindingtheGreatestCommonDivisor, ModularArithmetic, TheModulus, Linear Congruence, Properties of Congruences,Modular Arithmetic Operations, Modular Inverses,Properties of ModularArithmeti, Euclidean Algorithm Revisited, TheExtendedEuclideanAlgorithm, PrimeNumbers, Fermat's Theorem,Euler's Totient Function, Euler'sTheorem, TestingforPrimality, Miller–Rabin Algorithm, A Deterministic Primality Algorithm, Distribution of Primes, The Chinese Remainder Theorem, Discrete Logarithms, The Powers of an Integer ,Modulon , Logarithms for Modular Arithmetic, Calculation of Discrete Logarithms
[CO1]12 hrs

Unit 2:
Finite Fields : Algebraic Structures**,** Groups, Abelian Group, CyclicGroup, Rings, Fields, Finite Fields of the Form GF(*p*), FiniteFieldsofOrder*p*, Finding the Multiplicative Inverse in GF(*p*), Ordinary Polynomial Arithmetic, Polynomial Arithmetic with Coefficients in Z_p, Finding the Greatest Common Divisor, Finite Fields of the form **GF(2^n)**, Modular Polynomial Arithmetic,Finding the Multiplicative Inverse,Computational Considerations Using a Generator
CO2, 12 hrs

Unit 3:
Random Number Generation and Bitwise Operations: PrinciplesofPseudorandomNumberGeneration, PseudorandomNumberGenerators, True NumberGenerators, Binary Arithmetic, Bitwise AND, Bitwise OR, Bitwise XOR, Bitwise complement, Shift left, Shift right. [CO3].10 hrs

Unit 4:
Overview of Cryptography: Security terminology including Cryptology, Cryptography, Cryptanalysis, Confidentiality, Privacy, Threat, Attack, Incident, Intrusion, Malware, Countermeasure, Asset, Vulnerability, Risk, Mitigation of Risk, Cipher, Key, Symmetric Encryption, Asymmetric Encryption, Substitution and Transposition Ciphers, Block and Stream Ciphers. [C04] 10hrs.

Reference Books:

1. Cryptography & Network Security by Atul Kahate, Mc Graw Hill.
2. An introduction to mathematical Cryptography, Jeffrey Hoffstein, Jill Pipher, Joseph H., Springer.
3. Modern Cryptography: Applied Mathematics for Encryption and Information Security, William Easttom, Springer.

Course Outcomes:

Upon completing this course, the student will be able to

CO1	Apply different algorithms in number theory.
CO2	Apply finite fields in context to cryptography.
CO3	Implement random number generation.
CO4	Discuss overview of cryptography fundamentals.

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	**Course Title:** 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: BTCS 504-18	**Course Title:** Computer Networks	**3L:1T:0P**	**3Credits**	**42 Hours**

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.
3. TCP/IP Illustrated, Volume 1, W. Richard Stevens, Addison-Wesley, United States of America.

Course Code: BTCS507-18	**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: BTAI 502-20 Course Title : Artificial Intelligence 3L:0T:0P 3Credits

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: BTAI504-20 Course Title Artificial Intelligence Lab L:0;T:0;P:2 1Credits

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.

……

ELECTIVE-I

Course Name	**:**	**Web Technologies**
Course Code	**:**	**BTITCS503-20**
Credits	**:**	**3**
L T P	**:**	**3 0 2**

Course Objectives:

- To make students aware about Internet related technologies.
- To enable students to understand systematic way of developing a dynamic website by embedding HTML with various scripting languages like JavaScript, PHP etc.
- To develop an ability to choose best technologies for solving web client/server problems and designing a creative website.

Detailed Contents:

Module 1: Introduction: History and evolution of Internet protocols, Internet addressing, Internet Service Provider (ISP), Introduction to WWW, DNS, URL, HTTP, HTTPS, SSL, Web browsers, Cookies, Web servers, Proxy servers, Web applications, Website design principles, planning the site and navigation. [6 hrs][CO1]

Module 2: HTML and DHTML: Introduction, History of HTML, Structure of HTML Document: Text Basics, Structure of HTML Document: Images, Multimedia, Links, Audio, Video, Table and Forms, Document Layout, HTML vs. DHTML, Meta tags, Browser architecture and Website structure. [7 hrs][CO2]

Module 3: Style Sheets: Need for CSS, Introduction to CSS, Basic syntax and structure, Types of CSS – Inline, Internal and External style sheets.CSS Properties - Background images, Colors and properties, Text Formatting, Margin, Padding, Positioning etc., [5 hrs][CO2]

Module 4: Java Script: Introduction, JavaScript's history and versions, Basic syntax, Variables, Data types, Statements, Operators, Functions, Arrays, Objects, dialog boxes, JavaScript DOM. JQuery-Introduction, Installing & Configuration, jQuery Syntax, Selectors, Events, jQuery Callback & Chaining, Document Object Model, Validation [9 hrs][CO3]

Module 5: PHP and MySQL: Introduction and basic syntax of PHP, Data types, Variables,Decision and looping with examples, String, Functions, Array, Form processing, Cookies and Sessions, E-mail, PHP-MySQL: Connection to server. [7 hrs][CO4]

Module 6: Ajax and JSON: AJAX Introduction, AJAX Components, Handling Dynamic HTML with Ajax, Advantages & disadvantages, HTTP request, XMLHttpRequest Server Response .JSON– Syntax, Schema, Data types, Objects, Reading and writing JSON on client and server. Using JSON in AJAX applications. [8 hrs][CO5]

Course Outcomes: After studying this course the students will be able to-

CO1: Understand and apply the knowledge of web technology stack to deploy various web services.

CO2: Analyze and evaluate web technology components to design and implement static and dynamic website.

CO3: Apply JavaScript and related advance concepts for dynamic effects to create conforming web pages

CO4: Understand, analyze and build web applications using PHP with Database Connectivity.

CO5: Ability to design and deploy real world applications using appropriate client side and server side scripting languages

Suggested Books:

1. "Web Technologies: A Computer Science Perspective", Jeffrey C. Jackson, Pearson Education
2. "Internet and Web Technology" Rajkamal, , Tata McGraw Hill
3. "Web Enabled Commercial Application Development using HTML, DHTML JavaScript, Perl, CGI", Ivan Bayross, BPB Publications
4. "JavaScript JSON Cookbook", Ray Rischpater, "Packt Publishing
5. "PHP Black Book", Peter Moulding, Coriolis

Course Name	**:**	Web Technologies Lab
Course Code	**:**	BTITCS504-20
Credits	**:**	**1**
L T P	**:**	**0 0 2**

Course Objectives:

- Design and implement static and dynamic website using latest technical concepts
- Create conforming web pages by embedding HTML with various scripting languages like JavaScript, PHP etc.
- Develop an ability to choose best technologies for solving web client/server problems and designing a creative website

Experiment Task List:

1. Design a static web pages using HTML for online book store.
2. Design a dynamic web pages to demonstrate the usage of inline, internal and external cascading style sheets

3. Create a web page using XML.
4. Write a program to connect a XML web page to any database engine.
5. Write programs using Java script for Web Page to display browser's information.
6. Create a form with various fields and appropriate front and validations using any one of the scripting languages
7. Write a PHP program to keep track of the number of visitors visiting the web page and to display this count of visitors, with proper headings.
8. Write a PHP program to display a digital clock which displays the current time of the server.
9. Design a web applications using (a) PHP (b) Servlets (c) JSP
10. Database Connectivity with MySQL using Java Servlets, JSP, and PHP
11. Create a web application using React.js

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

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.

Course Code: BTITCS603-20	**Course Title:** Cyber Laws & IPR	3L:0T:0P	3 Credits

Detailed Contents:

Unit-I
Introduction

Overview of Computer and Web Technology, Need for Cyber Law, Cyber Jurisprudence at International and Indian Level, Jurisdictional Aspects in CyberLaw Issues of jurisdiction in cyberspace, Types of jurisdiction, Minimum Contacts Theory, Sliding Scale Theory, Effects Test and International targeting,Jurisdiction in IT Act, 2000. [CO1]

Unit-II

Cyber Crimes& Legal Framework

Cyber Crimes against Individuals, Institution and State, Hacking, Digital Forgery, Cyber Stalking/Harassment, Ethics and Etiquettes of Cyber World, Cyber Pornography, Identity Theft & Fraud, Cyber Terrorism, Cyber Defamation, Right to Privacy and Data Protection on Internet, Concept of privacy, Threat to privacy on internet, Self-regulation approach to privacy.[CO2]

Unit-III

Overview of Intellectual Property

introduction and the need for intellectual property right (IPR), IPR in India – Genesis and Development IPR in abroad, Data Protection, Open Source Software, Macro economic impact of the patent system, Patent and kind of inventions protected by a patent, Patent document How to protect your inventions?, Granting of patent, Rights of a patent.[CO3]

Unit-IV

Copyright, Related Rights and Trademarks

What is copyright? Latest editions of Designs, what is covered by copyright? How long does copyright last? Why protect copyright? What are related rights?,Distinction between related rights and copyright?, What is a trademark? Rightsof trademark?, What kind of signs can be used as trademarks?, types oftrademark, function does a trademark perform, How is a trademark protected?,

How is a trademark registered?[CO4][CO5]

Course Outcomes:

After completion of the course, students will able to:

1. Identify statutory, regulatory, constitutional, and organizational laws that affect the information technology professional.
2. Categorize case law and common law to current legal dilemmas in the technology field.
3. Outline the primary forms of intellectual property rights.
4. Compare the different forms of intellectual property protection in terms of their key differences and similarities.
5. Analyze the effects of intellectual property rights on society as a whole.

Text Books

1. Anirudh Rastogi. Cyber Law, LexisNexis.
2. Vakul Sharma. Information Technology Law and Practice Cyber Laws and Laws Relating to E-Commerce, Universal Law Publishing.
3. Pankaj Sharma. Information Security and Cyber Laws, Kataria, S. K., & Sons.
4. Navneet Nagpal. Intellectual Property Right, Ebooks2go Inc.
5. Dr. S.K. singh. Intellectual Property Rights, Central Law Agency.

Course Code: BTITCS604-20	**Course Title:** Cyber Laws & IPR Lab	0L:0T:2P	1Credits

List of Experiment:

1 Study of Jurisdictional Aspects in Cyber Law Issues
2 Study of Jurisdiction under IT Act, 2000.
3 Study of Hacking, Digital Forgery.
4 Study of threat to privacy on internet.
5 Study about the difference between related rights and copyright.
6 Study of Privacy and Data Protection on Internet.
7 Study about registration process of trademark.
8 Study about different kind of signs can be used as trademarks.
9 Study of Copyright, Related Rights and Trademarks.
10 Study of Self-regulation approach to privacy.
11 Study of intellectual property right (IPR) in India.
12 Study about impact of the patent system.
13 Study for Granting of patent.
14 Study related to Rights of Patents
15 Discussion with reference to authentication of Electronic Record using Digital Signatures

---

Sixth Semester

Course Code: BTCS 701-18	Course Title : Network Security and Cryptography	3L:0T:0P	3Credits

Detailed Contents:

UNIT 1: Introduction (3 Hours)

Introduction to Cryptography, Security Threats, Vulnerability, Active and Passive attacks, Security services and mechanism, Conventional Encryption Model, CIA model

[5hrs] (CO 1)

UNIT 2: Math Background

Modular Arithmetic, Euclidean and Extended Euclidean algorithm, Prime numbers, Fermat and Euler's Theorem

[5hrs] (CO 1)

UNIT 3: Cryptography

Dimensions of Cryptography, Classical Cryptographic Techniques Block Ciphers (DES, AES) : Feistal Cipher Structure, Simplifies DES, DES, Double and Triple DES, Block Cipher design Principles, AES, Modes of Operations Public-Key Cryptography : Principles Of Public-Key Cryptography, RSA Algorithm, Key Management, Diffie-Hellman Key Exchange, Elgamal Algorithm, Elliptic Curve Cryptography

[12hrs] (CO 2)

UNIT 4 Hash and MAC Algorithms

Authentication Requirement, Functions, Message Authentication Code, Hash Functions, Security Of Hash Functions And Macs, MD5 Message Digest Algorithm, Secure Hash Algorithm, Digital Signatures, Key Management : Key Distribution Techniques, Kerberos

[6hrs] (CO 3)

UNIT 5 Security in Networks

Threats in networks, Network Security Controls – Architecture, Encryption, Content Integrity, Strong Authentication, Access Controls, Wireless Security, Honeypots, Traffic flow security, Firewalls – Design and Types of Firewalls, Personal Firewalls, IDS, Email Security – PGP, S/MIME

[7hrs] (CO 4)

Course Outcomes:

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

CO1: Understand the fundamental principles of access control models and techniques, authentication and secure system design

CO2: Have a strong understanding of different cryptographic protocols and techniques and be able to use them.

CO3: Apply methods for authentication, access control, intrusion detection and prevention.

CO4: Identify and mitigate software security vulnerabilities in existing systems.

Suggested Readings/ Books:

1. Cryptography And Network Security Principles And Practice Fourth Edition, William Stallings, Pearson Education

2. Modern Cryptography: Theory and Practice, by Wenbo Mao, Prentice Hall PTR

3. Network Security Essentials: Applications and Standards, by William Stallings. Prentice Hall

4. Cryptography: Theory and Practice by Douglas R. Stinson, CRC press.

Course Code: BTITCS602-20	**Course Title:** Network Security and Cryptography Lab	0L:0T:2P	1 Credits

Lab as per the theory topics and tools designed by the instructor.

Course Code: BTITCS601-20	**Course Title:** Digital Forensics	3L:0T:0P	3 Credits

Type of course: Core

Prerequisite: Understanding of digital logic, operating system concepts, Computer hardware knowledge

Detailed Contents:

Module 1: Introduction
Understanding of forensic science, digital forensic, The digital forensic process, Locard's exchange principle, Scientific models.

Module 2: Understanding of the technical concepts
Basic computer organization, File system, Memory organization concept, Data storage concepts

Module 3: Digital Forensics Process Model
Introduction to cybercrime scene, Documenting the scene and evidence, maintaining the chain of custody, forensic cloning of evidence, Live and dead system forensic, Hashing concepts to maintain the integrity of evidence, Report drafting.

Module 4: Computer Operating system Artifacts
Finding deleted data, hibernating files, examining window registry, recycle bin operation, understanding of metadata, Restore points and shadow copies.

Module 5: Legal aspects of digital forensics
Understanding of legal aspects and their impact on digital forensics, Electronics discovery

Module 6: Understanding of digital Forensic tools
Quality assurance, Tool validation, Tool selection, Hardware and Software tools

Module 7: Case Study
Understanding of Internet resources, Web browser, Email header forensic, social networking sites

Course Outcomes:

After completion of the course, students will able to:

1. Describe Forensic science and Digital Forensic concepts.
2. Determine various digital forensic Operandi and motive behind cyber-attacks.
3. Interpret the cyber pieces of evidence, Digital forensic process model and their legal perspective.
4. Demonstrate various forensic tools to investigate the cybercrime and to identify the digital pieces of evidence.
5. Analyse the digital evidence used to commit cyber offences.

Reference Books:

1. The basics of digital Forensics (Latest Edition) – The primer for getting started in digital forensics by John Sammons – Elsevier Syngress Imprint.
2. Cybersecurity – Understanding of cybercrimes, computer forensics and Legal perspectives by Nina Godbole and Sunit Belapure – Wiley India Publication.
3. Practical Digital Forensics – Richard Boddington [PACKT] Publication, Open source community.

Course Code: BTITCS602-20	**Course Title:** Digital Forensics Lab	0L:0T:2P	1Credits

List of Experiment:

1. Study of computer forensics and different tools used for forensic investigation.
2. Live forensic case investigation using autopsy.
3. How to recover deleted files using forensic tools.
4. Find last connected USB on your system.
5. How to view last activity on your PC.
6. How to extracting browser Artifacts.
7. Comparison of two files for forensics investigation by comparing IT software.
8. How to collect e-mail evidence in victim PC.
9. Study the steps for hiding & extract any text file behind an image file/ audio file using command prompt.
10. How to extract exchangeable image file format (EXIF) data from image files using Exifreader software.

ELECTIVE-II

Course Code: BTITCS501-20	Course Title: Internet of Things	L:3; T:0; P:0	3Credits

DETAIL CONTENTS

1. Introduction to IoT

Architectural Overview, Design principles and needed capabilities, IoT Applications, Sensing, Actuation, Basics of Networking, M2M and IoT Technology Fundamentals- Devices and gateways, Data management, Business processes in IoT, Everything as a Service(XaaS), Role of Cloud in IoT, Security aspects in IoT.

(8 Hours) , CO1

2. Elements of IoT

Hardware Components- Computing (Arduino, Raspberry Pi), Communication, Sensing, Actuation, I/O interfaces. Software Components- Programming API's (using Python /Node.js /Arduino) for Communication, Protocols-MQTT, ZigBee, Bluetooth, CoAP, UDP, TCP

(9 Hours), CO2

3. IoT Application Development

Solution framework for IoT applications- Implementation of Device integration, Data acquisition and integration, Device data storage- Unstructured data storage on cloud/local server, Authentication, authorization of devices.

(18 Hours) CO3

4. IoT Case Studies

IoT case studies and mini projects based on Industrial automation, Transportation, Agriculture, Healthcare, Home Automation

(10 Hours),CO4

Course Outcomes: After the completion of this course, the students will be able to:

CO1: Understand internet of Things and its hardware and software components

CO2:Interface I/O devices, sensors & communication modules

CO3:Remotely monitor data and control devices

CO4:Develop real life IoT based projects

List of suggested books :

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

2. Dr. SRN Reddy, Rachit Thukral and Manasi Mishra, "Introduction to Internet of Things: A practical Approach", ETI Labs

3. Pethuru Raj and Anupama C. Raman, “The Internet of Things: Enabling Technologies, Platforms, and Use Cases”, CRC Press

4. Jeeva Jose, “Internet of Things”, Khanna Publishing House, Delhi

5. Adrian McEwen, “Designing the Internet of Things”, Wiley

6. Raj Kamal, “Internet of Things: Architecture and Design”, McGraw Hill

7. Cuno Pfister, “Getting Started with the Internet of Things”, O Reilly Media

Course Code: BTITCS505-20	**Course Title: Internet of Things Lab**	**L:0; T:0; P:2**	**1Credits**

LIST OF PRACTICALS

1. Familiarization with Arduino/Raspberry Pi and perform necessary software installation.

2. To interface LED/Buzzer with Arduino/Raspberry Pi and write a program to turn ON LED for 1 sec after every 2 seconds.

3. To interface Push button/Digital sensor (IR/LDR) with Arduino/Raspberry Pi and write a program to turn ON LED when push button is pressed or at sensor detection.

4. To interface DHT11 sensor with Arduino/Raspberry Pi and write a program to print temperature and humidity readings.

5. To interface motor using relay with Arduino/Raspberry Pi and write a program to turn ON motor when push button is pressed.

6. To interface OLED with Arduino/Raspberry Pi and write a program to print temperature and humidity readings on it.

7. To interface Bluetooth with Arduino/Raspberry Pi and write a program to send sensor data to smartphone using Bluetooth.

8. To interface Bluetooth with Arduino/Raspberry Pi and write a program to turn LED ON/OFF when ‘1’/’0’ is received from smartphone using Bluetooth.

9. Write a program on Arduino/Raspberry Pi to upload temperature and humidity data to thingspeak cloud.

10. Write a program on Arduino/Raspberry Pi to retrieve temperature and humidity data from thingspeak cloud.

11. To install MySQL database on Raspberry Pi and perform basic SQL queries.

12. Write a program on Arduino/Raspberry Pi to publish temperature data to MQTT broker.

Lab Outcomes: On successful completion of the course, the student will:

1.
2. Understand the concepts of Internet of Things
3. Analyze basic protocols in wireless sensor network
4. Design IoT applications in different domain and be able to analyze their performance
5. Implement basic IoT applications on embedded platform

Course Code: BTITCS703-20	**Course Title:** Ethical Hacking	3L:0T:0P	3Credits

Detailed Contents:

Module 1:

Ethical hacking process, Hacker's behaviour & mindset, Maintaining Anonymity, Hacking Methodology, Information Gathering, Active and Passive Sniffing, Physical security vulnerabilities and countermeasures. Internal and External testing. Preparation of Ethical Hacking and Penetration Test, Reports and Documents.

[8 hrs] (CO1)

Module 2:

Social Engineering attacks and countermeasures. Password attacks, Privilege Escalation and Executing Applications, Network Infrastructure Vulnerabilities, IP spoofing, DNS spoofing, Wireless Hacking: Wireless footprint, Wireless scanning and enumeration, Gaining access (hacking 802.11), WEP, WPA, WPA2.

[8 hrs] (CO2)

Module 3:

DoS attacks. Web server and application vulnerabilities, SQL injection attacks, Vulnerability Analysis and Reverse Engineering, Buffer overflow attacks. Client-side browser exploits, Exploiting Windows Access Control Model for Local Elevation Privilege. Exploiting vulnerabilities in Mobile Application.

[8 hrs] (CO3)

Module 4:

Introduction to Metasploit: Metasploit framework, Metasploit Console, Payloads, Metrpreter, Introduction to Armitage, Installing and using Kali Linux Distribution, Introduction to penetration testing tools in Kali Linux. Case Studies of recent vulnerabilities and attacks.

[8 hrs] (CO4)

Module 5:

Malware Analysis: Collecting Malware and Initial Analysis, Hacking Malware, Case study of vulnerability of cloud platforms and mobile platforms & devices.

[10 hrs] (CO5)

Course Outcomes:

The student will be able to:

1. Learn about different tools and techniques in Ethical hacking and security.

2. Understand ethics behind hacking and vulnerability disclosure.

3. Understand the core concepts related to vulnerabilities and their causes.

4. Exploit the vulnerabilities related to computer system and networks using state of the art tools and technologies.

5. Ethical hacking tools to perform various activities.

Reference Books:

1. Shon Harris, Allen Harper, Chris Eagle and Jonathan Ness, Gray Hat Hacking: The Ethical Hackers' Handbook, TMH Edition

2. Jon Erickson, Hacking: The Art of Exploitation, SPD

3. Baloch, R., Ethical Hacking and Penetration Testing Guide, CRC Press, 2015.

4. Beaver, K., Hacking for Dummies, 3rded. John Wiley & sons., 2013.

5. Council, Ec. , Computer Forensics: Investigating Network Intrusions and Cybercrime, Cengage Learning, Second Edition, 2010

6. McClure S., Scambray J., and Kurtz G, Hacking Exposed. Tata McGraw-Hill Education, 6th Edition, 2009 5. International Council of E-Commerce Consultants by Learning, Penetration Testing Network and Perimeter Testing Ec-Council/ Certified Security Analyst Vol. 3 of Penetration Testing, Cenage Learning, 2010

7. Davidoff, S. and Ham, J., Network Forensics Tracking Hackers through Cyberspace, Prentice Hall, 2012. 7. Michael G. Solomon, K Rudolph, Ed Tittel, Broom N., and Barrett, D.,Computer, Forensics Jump Start, Willey Publishing, Inc, 2011.

Course Code: BTITCS704-20	**Course Title:** Ethical Hacking Lab	0L:0T:4P	2Credits

List of Experiments:

Task 1: Setup a honey pot and monitor the honey pot on network.

Task 2: Write a script or code to demonstrate SQL injection attacks.

Task 3: Create a social networking website login page using phishing techniques.

Task 4: Write a code to demonstrate DoS attacks.

Task 5: Install rootkits and study variety of options.

Task 6: Study of Techniques uses for Web Based Password Capturing.

Task 7: Install jcrypt tool (or any other equivalent) and demonstrate Asymmetric, Symmetric Crypto algorithm, Hash and Digital/PKI signatures studied in theory Network Security And Management.

Task 8: Implement Passive scanning, active scanning, session hijacking, cookies extraction using Burp suit tool.

Lab Outcomes:

The student will be able to:

1. The aim of the course is to introduce the methodologies framework tools of ethical hacking to get awareness in enhancing the security

2. To get knowledge on various attacks and their detection.

3. Gain the knowledge of the use and availability of tools to support an ethical hack.

4. Gain the knowledge of interpreting the results of a controlled attack.

Reference Books:

1. Shon Harris, Allen Harper, Chris Eagle and Jonathan Ness, Gray Hat Hacking: The Ethical Hackers' Handbook, TMH Edition

2. Jon Erickson, Hacking: The Art of Exploitation, SPD

3. Baloch, R., Ethical Hacking and Penetration Testing Guide, CRC Press, 2015.

4. Beaver, K., Hacking for Dummies, 3rded. John Wiley & sons., 2013.

Course Code: BTITCS707-20	**Course Title:** Cyber Attacks	3L:0T:0P	3 Credits

Module 1: Introduction to Cyber Security & Attacks

Introduction, Computer Security, Threats, Harm, Vulnerabilities, Controls, Authentication, Access Control and Cryptography

Web attack: Browser Attacks, Web Attacks Targeting Users, Obtaining User or Website Data, Email Attacks.

Network Vulnerabilities: Overview of vulnerability scanning, Open Port / Service Identification, Banner/Version Check, Traffic Probe, Vulnerability Probe, Vulnerability Examples, OpenVAS, Metasploit.

Networks Vulnerability Scanning (Netcat, Socat), Network Sniffers and Injection tools [CO 1,2]

Module 2: Network Defence tools.

Firewalls and Packet Filters: Firewall Basics, Packet Filter Vs Firewall, How a Firewall Protects a Network, Packet Characteristic to Filter, Stateless Vs Stateful Firewalls, Network Address Translation (NAT) and Port Forwarding.

VPN: the basic of Virtual Private Networks.

Firewall: Introduction, Linux Firewall, Windows Firewall.

Snort: Introduction Detection System. [CO 1,3]

Module 3: Web Application Tools

Scanning for web vulnerabilities tools: Nikto, W3af

HTTP utilities - Curl, OpenSSL and Stunnel.

Application Inspection tools – Zed Attack Proxy, Sqlmap, DVWA, Webgoat.

Password Cracking and Brute-Force Tools: John the Ripper, L0htcrack, Pwdump, HTC-Hydra. [CO 1,2]

Module 4: Introduction to Cyber Attack, law and Investigation

Hacking, Attack vectors, Cyberspace and Criminal Behaviour, Digital Forensics, Computer Language, Network Language, Realms of the Cyber world.

Firewalls and Packet Filters, password Cracking, Keyloggers and Spyware, Virus and Warms, Trojan and backdoors, Steganography, DOS and DDOS attack, SQL injection, Buffer Overflow, Attack on wireless Networks. [CO 3,4]

Course Outcomes:

After completion of the course, students will able to:

1. Analyze and resolve security issues in networks and computer systems to secure an IT infrastructure.
2. Design, develop, test and evaluate secure software.
3. Develop policies and procedures to manage enterprise security risks and attacks.
4. Interpret and forensically investigate security incidents.

Reference Books:

1. Margaret J. Goldstein - Martin Gitlin "Cyber Attack", Twenty-First Century Books, 2015
2. Paul Day, "Cyber Attack: The Truth about Digital Crime, Cyber Warfare and Government Snooping", Carlton books limited, 2014.
3. Edward Amoroso, "Cyber Attacks: Protecting National Infrastructure", Elsevier, 2012.

Course Code: BTITCS708-20	**Course Title:** Cyber Attacks	0L:0T:2P	1 Credits

Lab may be designed by instructor based on theory curriculum.

ELECTIVE-III

Course Code: BTCS 618-18	Course Title : Machine Learning	3L:0T:0P	3Credits

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: BTCS619-18	**Course Title: Machine Learning Lab**	**L:0;T:0;2 P:**	**1Credits**

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

Lab Outcomes: After successful completion of the lab, students can able to:

1. Solve problems using the machine learning models.
2. Apply various reinforcement algorithms to solve real time complex problems.
3. Identify the core components of deep neural network model.
4. Implement unsupervised models through programming language.

Course Code: BTITCS605-20	**Course Title:** Intrusion Detection System	3L:0T:0P	3Credits

Pre-requisites: Fundamental knowledge in Operating Systems, and Networks

Detailed Contents:

Module 1:

History of Intrusion detection, Audit, Concept and definition , Internal and external threats to data, attacks, Need and types of IDS, Information sources Host based information sources, Network based information sources.

Module 2:

Intrusion Prevention Systems, Network IDs protocol based IDs, Hybrid IDs, Analysis schemes, thinking about intrusion. A model for intrusion analysis, techniques Responses requirement of responses, types of responses mapping responses to policy Vulnerability analysis, credential analysis non credential analysis

Module 3:

Introduction to Snort, Snort Installation Scenarios, Installing Snort, Running Snort on Multiple Network Interfaces, Snort Command Line Options. Step-By-Step Procedure to Compile and Install Snort Location of Snort Files, Snort Modes Snort Alert Modes

Module 4:

Working with Snort Rules, Rule Headers, Rule Options, The Snort Configuration File etc. Plugins, Preprocessors and Output Modules, Using Snort with MySQL

Module 5:

Using ACID and Snort Snarf with Snort, Agent development for intrusion detection, Architecture models of IDs and IPs.

Course Outcomes:

The student will be able to:

1. Explain the fundamental concepts of Network Protocol Analysis and demonstrate the skill to capture and analyse network packets.
2. Use various protocol analysers and Network Intrusion Detection Systems as security tools to detect network attacks and troubleshoot network problems.

Suggested Books:

1. Rafeeq Rehman : " Intrusion Detection with SNORT, Apache, MySQL, PHP and ACID," 1st Edition, Prentice Hall , 2003.

Reference Books:

1. Christopher Kruegel,Fredrik Valeur, Giovanni Vigna: "Intrusion Detection and Correlation Challenges and Solutions", 1st Edition, Springer, 2005.
2. Carl Endorf, Eugene Schultz and Jim Mellander " Intrusion Detection & Prevention", 1st Edition, Tata McGraw-Hill, 2004.
3. Stephen Northcutt, Judy Novak : "Network Intrusion Detection", 3rd Edition, New Riders Publishing, 2002.
4. T. Fahringer, R. Prodan, "A Text book on Grid Application Development and Computing Environment". 6th Edition, KhannaPublihsers, 2012.

Course Code: BTITCS606-20	**Course Title:** Intrusion Detection System Lab	0L:0T:2P	1Credits

Lab as per the theory topics and tools designed by the instructor.

Course Code: BTITCS705-20	**Course Title:** Vulnerability Analysis & Penetration testing	3L:0T:0P	3Credits

UNIT 1:

INTRODUCTION: Penetration Testing phases/Testing Process, types and Techniques, Blue/Red Teaming, Strategies of Testing, Non-Disclosure Agreement Checklist, Phases of hacking, Open-source/proprietary Pentest Methodologies

UNIT 2:

INFORMATION GATHERING AND SCANNING:

Information gathering methodologies- Foot printing, Competitive Intelligence DNS Enumerations- Social Engineering attacks, Port Scanning, Network Scanning, Vulnerability Scanning, NMAP scanning tool, OS Fingerprinting, Enumeration

UNIT 3:

SYSTEM HACKING

Password cracking techniques- Key loggers- Escalating privileges- Hiding Files, Double Encoding, Steganography technologies and its Countermeasures. Active and passive sniffing- ARP Poisoning, MAC Flooding- SQL Injection – Error based, Union-based, Time-based, Blind SQL, Out-of-band. Injection Prevention Techniques.

UNIT 4:

ADVANCED SYSTEM HACKING

Broken Authentication, Sensitive Data Exposure, XML External Entities, Broken Access Code, XSS - Stored, Reflected, DOM Based

UNIT 5;

WIRELESS PENTEST

Wi-Fi Authentication Modes, Bypassing WLAN Authentication, Types of Wireless Encryption, WLAN Encryption Flaws, AP Attack, Attacks on the WLAN Infrastructure, DoS-Layer1, Layer2, Layer 3, DDoS Attack, Client Misassociation, Wireless Hacking Methodology, Wireless Traffic Analysis

REFERENCE BOOKS

1 Kali Linux Wireless Penetration Testing Beginner's Guide by Vivek Ramachandran, Cameron Buchanan, 2015 Packt Publishing

2 SQL Injection Attacks and Defense 1st Edition, by Justin Clarke-Salt, Syngress Publication

3 Mastering Modern Web Penetration Testing By Prakhar Prasad, October 2016 Packt Publishing

4 Kali Linux 2: Windows Penetration Testing, By Wolf Halton, Bo Weaver , June 2016 Packt Publishing

Course Code: BTITCS705-20	**Course Title:** Vulnerability Analysis & Penetration testing LAb	0L:0T:4P	2Credits

List of practical:

1. To explore vulnerability scanning and its types including network, port and vulnerability scanning
2. Use Nmap tool for live scanning on ports and networks
3. Netcat usage on TCP/UDP ports
4. Wireshark basics and capturing data on LAN using this sniffing tool
5. Perform NFS ,SMB ,SMTP enumeration
6. Nessus installation and configuration
7. Perform Vulnerability scanning with Nessus
8. Web application assessment with nikto & burp suite
9. Vulnerability analysis with Metasploit framework
10. To perform penetration testing using KALI linux tools on dummy websites
11. Any other practical as per requirement of understanding concepts outlined in theory syllabus

Scheme & Syllabus of

Bachelor of Technology

Computer Science & Engg.

(Cyber Security)

Batch 2021 onwards
(7th/ 8th Semester)

By

Department of Academics

IK Gujral Punjab Technical University

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
BTIT CS701 -20	Professional Core Courses	Blockchain & Cryptocurrency	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 604-18	Professional Core Courses	Compiler Design Lab	0	0	2	30	20	50	1
BTIT CS701 -20	Professional Core Courses	Blockchain & Cryptocurrency Lab	0	0	2	30	20	50	1
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**	**20**	**440**	**460**	**900**	**25**

Seventh/Eighth Semester

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

*Students may be encouraged to acquire some professional certification during this like EC Council of India certified Hacker/ PT etc.

LIST OF ELECTIVES

Elective IV
BTITCS709-20 Applied Cryptography
BTITCS710-20 Applied Cryptography Lab
BTITCS608-20 IoT Security
BTITCS609-20 IoT Security Lab
BTITCS701-20 Distributed Computing
BTITCS702-20 Distributed Computing Lab

Elective V
BTAIML709-20 Applied Intelligence
BTAIML710-20 Applied Intelligence lab
BTAIML603-20 Neural Networks
BTAIML604-20 Neural Networks Lab
BTIT703-20 Attacks in IOT N/W
BTIT704-20 Attacks in IOT N/W Lab

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, IntermediateLanguages: Syntax Tree, Three Address Code, Types and Declarations, Translation ofExpressions, 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: Understand 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: BTITCS701-20	Course Title: Block chain & Cryptocurrency	3L:0T:0P	3Credits

Detailed Contents:

Module 1: Introduction to Blockchain

Introduction: Overview of Block chain, Public Ledgers, Bitcoin, Smart Contracts, Block in a Block chain, Transactions, Distributed Consensus, Public vs Private Block chain, Understanding Cryptocurrency to Block chain, Permissioned Model of Block chain, Overview of Security aspects of Block chain.

[8 hrs] (CO1)

Module 2: Basic Crypto Primitives:

Primitives: Cryptographic Hash Function, Properties of a hash function, Hash pointer and Merkletree, Digital Signature, Public Key Cryptography, Private Key Cryptography, A basic cryptocurrency.

[8 hrs] (CO2, CO3)

Module 3: Understanding Blockchain with Crypto currency

Bitcoin and Block chain: Creation of coins, Payments and double spending, Bitcoin Scripts, Bitcoin P2P Network, Transaction in Bitcoin Network, Block Mining, Block propagation andblock relay.

[8 hrs] (CO3)

Module 4: Understanding Blockchain for Enterprises

Enterprise application of Block chain: Cross border payments, Know Your Customer (KYC), Food Security, Mortgage over Block chain, Block chain enabled Trade, We Trade –Trade Finance Network, Supply Chain Financing, Identity on Block chain.

[8 hrs] (CO4, CO5)

Module 5: Blockchain application development

Hyperledger Fabric- Architecture, Identities and Policies, Membership and Access Control, Channels, Transaction Validation, Writing smart contract using Hyperledger Fabric, Writingsmart contract using Ethereum, Overview of Ripple and Corda.

[8 hrs] (CO4, CO5)

Module 6: Recent Trends and Technology in blockchain

Blockchain as a Service (BaaS) By Big Tech Companies. One of the promising blockchain trends in 2020 is BaaS, short for Blockchain As A Service. It is a new blockchain trend that is currently integrated with a number of startups as well as enterprises.

Course Outcomes:

The student will be able to:

1 : Understand block chain technology [8 hrs] (CO5,CO6

2: Develop block chain based solutions and write smart contract using Hyperledger Fabric andEthereum frameworks

3: Build and deploy block chain application for on premise and cloud based architecture.

4: Integrate ideas from various domains and implement them using block chain technology indifferent perspectives.

5: Able to develop blockchain applications

6: Understand the security Features in blockchain technology and develop applications

Suggested Books:

1. Kalle Rosenbaum, *Grokking Bitcoin*, MANNING Publication.
2. Lorne Lantz & Daniel Cawrey, *Mastering Blockchain Unlocking the Power of Cryptocurrencies, Smart Contracts, and Decentralized Applications*, O'REILLYPublications.
3. The Basics of Bitcoins and Blockchains: An Introduction to Cryptocurrencies and the Technology that Powers Them (Cryptography, Derivatives Investments, Futures Trading,Digital Assets, NFT) Hardcover – Illustrated, September 15, 2018.

Reference Books:

1. Bina Ramamurthy, *Blockchain in Action*, MANNING Publication
2. Bikramaditya Singhal, Gautam Dhameja, and Priyansu Sekhra Panda, *BeginningBlockchain*, Apress Publication

Course Code: BTITCS702-20	**Course Title:** Block chain & Cryptocurrency Lab	0L:0T:2P	1Credits

List of Experiment:

Task1: WAP to implement Merkle Tree.
Task 2: WAP for Creation of block.
Task 3: WAP for Blockchain programming code.
Task 4: WAP to implement ERC20 Token.
Task 5: Implementation of a blockchain in C++ - Peer-to-Peer network, SHA-256, Merkle Trees,Mining
Task 6: Implementation of Data Encryption Standard (DES)
Task 7: Implementation of MD5.
Task 8:Implementation of SHA-1.
Task9.Implement the Signature Scheme for Digital Signature Standard.

Lab Outcomes:

The student will be able to:
1.Implement the cipher techniques.
2.Develop the various security algorithms.
3.Use different open source tools for network security and analysis

Course Code: BTITCS709-20	**Course Title:** Applied Cryptography	3L:0T:0P	3Credits

Detailed Contents:

Module 1:

History of cryptography, some background in probability and algorithms, classical cryptography (shift cipher, monoalphabetic substitution cipher, polyalphabetic substitution cipher), encryption with perfect secrecy, one-time pad.

Some background in algorithms and complexity theory, modern cryptography principles, one-way functions, trapdoor functions, hard-core bits, construction of a public-key cryptosystem based on general cryptographic primitives.

[6 hrs] (CO1)

Module 2:

Algorithmic number theory, number theory and cryptographic assumptions, Reductions, proofs by reductions, number theory candidates for cryptographic primitives (e.g., factoring and related problems), public-key cryptosystems from number theory problems; brief discussion of quantum computing.

Randomness and pseudo-randomness, pseudo-random generators, functions and permutations. Symmetric encryption: introduction, security notions, symmetric encryption schemes based on pseudo-randomness primitives, security proofs, fundamental concepts.

[6 hrs] (CO2)

Module 3:

Symmetric encryption: block ciphers (e.g., DES, Triple-DES, AES), substitution/permutation networks, Feistel networks, modes of operations (e.g., ECB, CBC, OFB, Counter), cryptanalysis attacks (e.g., exhaustive, linear, differential, meet-in-the-middle attack), key lengths.

Message authentication: introduction, notion and schemes (e.g., CBC-MAC), collision resistant hashing (MD5, SHA-1, SHA-2, SHA-3, HMAC, Merkle-Hellman), CCA security for symmetric encryption, simultaneous message confidentiality and message integrity, the GCM mode, application case study 1: password-based secure computer access.

[8 hrs] (CO2)

Module 4:

More number theory candidates for cryptographic primitives (e.g., discrete logarithms, brief discussion of related problems including elliptic curves). Asymmetric encryption: comparison with symmetric encryption, definitions, constructions (e.g., RSA variants, El Gamal), hybrid encryption; implementation aspects: security-performance-features trade-offs.

Asymmetric encryption: malleable and homomorphic encryption notion and schemes (e.g., Paillier, brief discussion of various schemes, including Gentry's), additional schemes achieving various security notions in various models (e.g., Cramer-Shoup), identity-based encryption.

[8 hrs] (CO2, CO3)

Module 5:

Property-preserving public-key encryption, secure 2-party computation, secure multi party computation; application case study 2: sugar beet auction; implementation aspects of cryptographic protocols: transport layer, protocols over secure channels.

Digital Signatures, hashing and signing, Hashed RSA, El Gamal and DSA signature schemes, public-key infrastructures, certificates, cryptography in TLS, IPSec and virtual private networks, NSA Suite B, application case study 3: secure online purchasing; implementation aspects: trust models, PKI implementation challenges.

[8 hrs] (CO2,CO3)

Module 6:

Key protocols: key transport, key agreement, notions and schemes (e.g., Diffie-Hellman schemes); key management: concepts and lifecycle; code obfuscation, application case study 4: digital rights management; quantum computing, quantum-resistant cryptography; implementation aspects: creating correct and secureprograms, quality of code, side-channel attacks, implementation flaws.

Key lengths and recommendations, user authentication: password, challenge-response and zero-knowledgeprotocols; server authentication; application case study 5: secure online banking; digital cash, application case study 6: keeping/storing secrets, blockchain, application case study 7: cryptocurrencies; implementation aspects: weakest key, key modularity.

[8 hrs] (CO4)

Course Outcomes:

The student will be able to:

1. Learn the main areas of Modern Cryptography, including their main problem statements and the rigorous mathematical approaches used to formalize them
2. Learn and describe how various cryptographic algorithms and protocols work, pointing out the main techniques used in them, and proving/disproving most basic properties, such as correctness of decryption, digital signatures, authentication tags, and key agreement
3. Evaluate functionality, security and performance properties of cryptography methods used as components of complex security solutions
4. Analyze the impact of errors or different designs of cryptography algorithms and protocols 5. Describe the applications of cryptography algorithms and protocols to real-life problems and manyimplementation issues in developing these solutions.

Suggested Books:

1. N. Ferguson, B. Schneier and T. Kohno, Cryptography Engineering: Design, Principles andPractical Applications, Wiley Publishing, Inc., 2010.
2. B. Schneier, Applied Cryptography, 2nd edition, J. Wiley and Sons.

Reference Books:

1. W. Stallings, Cryptography and Network Security: Principles and Practice, 2nd edition, PrenticeHall
2. Books at http://www.freetechbooks.com/information-security-f52.html

Course Code: BTITCS710-20	**Course Title:** Applied Cryptography Lab	0L:0T:2P	1Credits

All the algorithms mentioned in theory may be implemented using C/ C++/ python.

Course Code: BTITCS608-20	**Course Title:** IoT Security	3L:0T:0P	3 Credits

Module 1:

IoT-Architecture, Functional-Architecture, Layered model, Phases of IoT system, Internet of Things Attack surface, Applied Physical Attacks-Recon and Passive Analysis, Recognizing and communicating hardware impact, Sourcing documentation and tools, Reading datasheets andinferring system functionality

Module 2:

Threat Modeling and System Analysis, Threat modeling when hardware is in scope, Dynamic analysis, Analyzing interconnects, Analyzing an unknown protocol, Firmware vulnerability analysis and exploitation. Static vs Dynamic analysis and tools, Dynamic analysis in-circuit vsemulator, Tooling for dynamic analysis

Module 3:

Trust and security from a device perspective, Secure key storage, Trust and security from a network perspective, PKI Architecture Components, A Public Key Reference Infrastructure forthe Iot.

Module 4:

Characterizing Complex Systems: Wireless networks, Biological networks, Social networks, Economic networks, Computer networks. Computational Tools for Complex Systems: Signal processing tools, Network science tools, Controllability and observability of networks, Network tomography, Lessons from communications engineering.

Course Outcomes:

After completion of the course, students will able to:

1. Understanding IoT Architectures and Attack surface.
2. Learn Recon and Passive Analysis on Hardware Layer.
3. Learn Threat Modelling and System Analysis.
4. Learn Firmware Vulnerability Analysis and Exploitation.

Reference Books:

1. Fei HU, "Security and Privacy in Internet of Things (IoTs): Models, Algorithms, and Implementations", CRC Press,2016
2. Russell, Brian and Drew Van Duren, "Practical Internet of Things Security", Packet Publishing, 2016.
3. Ollie Whitehouse, "Security of Things: An Implementers' Guide to Cyber-Security for Internet of Things Devices and Beyond", NCC Group, 2014

Course Code: BTITCS609-20	**Course Title:** IoT Security Lab	**0L: 0T: 2P**	**Credits: 1**

Lab as per the theory topics and tools designed by the instructor.

Course Code: BTITCS 701-20	**Course Title: Distributed Computing**	3L:0T:0P	3 Credits

Detailed Contents:

UNIT I: Introduction to Computer System Components, Message -Passing Systems versus Shared Memory Systems, Primitives for Distributed Communication, Synchronous versus Asynchronous Executions. Design Issues and Challenges of Distributed Computations. Models of Communication Networks, Global State of a Distributed System.

UNIT II: Logical Time: Physical Clock Synchronization: NTP – A Framework for a System of Logical Clocks, Scalar Time and Vector Time; Message Ordering and Group Communication: Message Ordering Paradigms – Asynchronous Execution with Synchronous Communication – Synchronous Program Order on Asynchronous System, Group Communication

UNIT III: Distributed Mutual exclusion Algorithms: Introduction and Preliminaries. Lamport's algorithm, Ricart Agrawala's Algorithm, Token-Based Algorithms, Suzuki-Kasami's Broadcast Algorithm; Deadlock and its Detection in Distributed Systems, Deadlock Models

UNIT IV: Consensus and Agreement Algorithms: Problem Definition – Overview of Results – Agreement in a Failure-Free System (Synchronous and Asynchronous), Agreement in Synchronous Systems with Failures; Checkpointing and Rollback Recovery: Introduction – Background and Definitions – Issues in Failure Recovery. Checkpoint-based Recovery, Coordinated Checkpointing Algorithm, Algorithm for Asynchronous Checkpointing and Recovery

UNIT V: Definition of Cloud Computing, Characteristics of Cloud, Cloud Deployment Models, Cloud Service Models, Driving Factors and Challenges of Cloud, Virtualization, Load Balancing, Scalability and Elasticity, Replication, Cloud Services and Platforms.

References:

1. Kshemkalyani Ajay D, Mukesh Singhal, "Distributed Computing: Principles, Algorithms and Systems", Cambridge Press, 2011.
2. Mukesh Singhal, Niranjan G Shivaratri, "Advanced Concepts in Operating systems", McGraw Hill Publishers, 1994.

Course Code: BTITCS 702-20	**Course Title: Distributed Computing Lab**	0L:0T:2P	Credit 1

Lab as per the theory topics and tools designed by the instructor.

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 aerospacesector 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 andcase 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 RecurrentNeural Networks. [10 hrs] (CO5)

Course Outcomes:
The student will be able to:
1. Explain fundamental meaning and discuss applicability of machine learning algorithms forindustrial applications.
2. Test the commonly used AI algorithms and describe their applications.
3. Implement AI algorithms, estimate their performance in a simulation environment andassess their performance for a realistic case study.
4. Judge AI implementation platforms and create deep learning applications for specificproblems.
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

Course Code: BTAIML710-20	**Course Title:** Applied Intelligence Lab	0L:0T:2P	1Credits

List of Experiment:

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 of any health sector dataset
4. Write a programme to infer from the Bayesian network on the above dataset
5. Write a programme to run value and policy iteration in a grid world in real world problem
6. Write a programme to do reinforcement learning in a grid world in real world problem

Lab Outcomes:

Upon successful completion of the course, the student will be able toCO1 Apply various pre-processing techniques on different datasets.
CO2 Construct Machine learning programs for Supervised, Unsupervised and Semisupervised learning models.
CO3 Develop Deep learning programs for Supervised & Unsupervised learning models.CO4 Identify and Apply Applied Intelligence concepts to solve real world problems.

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

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, Learningwithout 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 ofback 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, solvingsimultaneous 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 multilayerperceptrons, 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 featurespace.
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: BTIT703-20	Course Title: Attacks in IOT Networks	3L:0 T: 0P	Credits: 3

Credit: 3

Unit 1: Introduction to Operational Technology, Overview of industrial control systems (ICS), ICS operation & components, Cyber-physical systems (CPS) & IoT

Unit 2: IoT Vulnerabilities, Threats & Risks: STRIDE methodology, OWASP Iot vulnerabilities, Privacy & trust, Insufficient authentication/authorization, Insufficient access control, Attacks on IoT data, Attacks on IoT layered architecture, Security concerns in IoT applications, Security concerns in SCADA

Unit 3: IoT Pen testing: Active vulnerability analysis tools, Port scanning, Operating system fingerprinting and version scanning, Penetration testing, Attack surface mapping

Unit 4: Firmware Reverse Engineering: Understanding firmware, Extracting firmware, Manual firmware extraction, Automated file system extraction, Firmware internals, Backdooring a firmware, Static & dynamic analysis

Unit 5: Radio & Side Channel Attacks, Software defined radio, Exploiting ZIGBEE & BLE, Power analysis attack, Invasive attack, Perturbation -attacks, Electromagnetic side channel attack, fault injection attack, timing attack, covert channel attacks

Reference Books:

1 “Securing the Internet of Things”, Shancang Li, Li Da Xu, Syngress,Elsevier, 2017
2 “Security and Privacy in Internet of Things (IoTs)Models, Algorithms, and Implementations”, Edited by Fei Hu, CRC Press, 2016
3 “Practical Internet of Things Security”, Brian Russell Drew Van Duren, Packt Publishing, 2016

Course Code: BTIT704-20	Course Title: Attacks in IOT Networks Lab	0L:0 T: 2P	Credits: 1

Lab as per the theory topics and tools designed by the instructor.