3rd Semester		Contact Hrs.			Marks			Credits
Code	**Course**	**L**	**T**	**P**	**Int.**	**Ext.**	**Total**	
BITE2-302	Data Structures	3	1	0	40	60	100	4
BITE2-303	Object Oriented Programming Using C++	3	0	0	40	60	100	3
BITE2-304	Digital Circuits & Logical Design	3	0	0	40	60	100	3
BITE2-305	Computer Architecture & Organization	3	0	0	40	60	100	3
BITE2-306	Discrete Structures	3	1	0	40	60	100	4
BITE2-307	Data Structures Lab.	0	0	2	60	40	100	1
BITE2-308	Object Oriented Programming Using C++ Lab.	0	0	2	60	40	100	1
BITE2-309	Digital Circuit & Logical Design Lab.	0	0	2	60	40	100	1
BHUM0-F91	Soft Skills-I	0	0	2	60	40	100	1
BITE2-310	Training-I#	0	0	4	60	40	100	2
Total		**15**	**2**	**12**	**500**	**500**	**1000**	**23**

 # 4-Week Training during summer vacations after 2nd semester

4th Semester		Contact Hrs.			Marks			Credits
Code	**Course**	**L**	**T**	**P**	**Int.**	**Ext.**	**Total**	
BITE2-411	Operating System	3	0	0	40	60	100	3
BITE2-412	Database Management Systems-I	3	0	0	40	60	100	3
BITE2-413	Computer Networks-I	3	0	0	40	60	100	3
BITE2-414	Design & Analysis Of Algorithms	3	1	0	40	60	100	4
BITE2-415	Microprocessors & Assembly Languages	3	0	0	40	60	100	3
BITE2-416	Database Management Systems-I Lab.	0	0	4	60	40	100	2
BITE2-417	Computer Networks-I Lab.	0	0	2	60	40	100	1
BITE2-418	Design & Analysis of Algorithms Lab.	0	0	2	60	40	100	1
BITE2-419	Microprocessors & Assembly Languages Lab.	0	0	2	60	40	100	1
BHUM0-F92	Soft Skills-II	0	0	2	60	40	100	1
Total		**15**	**1**	**12**	**500**	**500**	**1000**	**22**

5th Semester		Contact Hrs.			Marks			Credits
Code	**Course**	**L**	**T**	**P**	**Int.**	**Ext.**	**Total**	
BITE2-520	System Analysis and Design	3	1	0	40	60	100	4
BITE2-521	Programming in Java	3	1	0	40	60	100	4
BITE2-522	Computer Networks-II	3	1	0	40	60	100	4
Departmental Electives-I (Choose any one)		3	1	0	40	60	100	4
BITE2-556	Cyber Laws & IPR							
BITE2-557	Compute4r Graphics							
BITE2-558	Linus & Shell Programming							
BITE2-523	Programming in Java Lab.	0	0	2	60	40	100	1
BITE2-524	Computer Networks-II Lab.	0	0	2	60	40	100	1
BHUM0-F93	Soft Skills-III	0	0	2	60	40	100	1
Open Elective-I (Choose any one)		3	0	0	40	60	100	3
BITE2-525	Training-II#	0	0	4	60	40	100	2
Total		**15**	**4**	**10**	**440**	**460**	**900**	**24**

 # 6-Week Training during summer vacations after 4th semester

6th Semester		Contact Hrs.			Marks			Credits
Code	**Course**	**L**	**T**	**P**	**Int.**	**Ext.**	**Total**	
BITE2-626	Network Programming	3	1	0	40	60	100	4
BITE2-627	Software Engineering	3	1	0	40	60	100	4
Departmental Electives-II (Choose any one)		3	1	0	40	60	100	4
BITE2-659	Mobile App Development							
BITE2-660	Cryptography & Network Security							
BITE2-661	Web Technologies							
Departmental Electives-III (Choose any one)		3	1	0	40	60	100	4
BITE2-662	Cloud Computing							
BITE2-663	Enterprise Resource Planning							
BITE2-664	Parallel & Distributed							
BITE2-628	Network Programming Lab.	0	0	2	60	40	100	1
BITE2-629	Software Engineering Lab.	0	0	2	60	40	100	1
Open Elective-II (Choose any one)		3	0	0	40	60	100	3
BHUM0-F94	Soft Skills-IV	0	0	2	60	40	100	1
Total		**15**	**4**	**6**	**380**	**420**	**800**	**22**

7th Semester		Contact Hrs.			Marks			Credits
Code	**Course**	**L**	**T**	**P**	**Int.**	**Ext.**	**Total**	
BITE2-730	Building Enterprise Applications	3	1	0	40	60	100	4
BITE2-731	Simulation and Modeling	3	1	0	40	60	100	4
BITE2-732	Building Enterprise Applications Lab.	0	0	4	60	40	100	2
BITE2-733	Simulation and Modeling Lab.	0	0	4	60	40	100	2
BITE2-734	Training-III#	0	0	0	60	40	100	4
BITE2-735	Project-I	0	0	8	60	40	100	4
Departmental Electives-IV (Choose any one)		3	1	0	40	60	100	4
BITE2-765	Information Security and Risk Management							
BITE2-766	Digital Image Processing							
BITE2-767	Software Project Management							
Total		**9**	**3**	**16**	**360**	**340**	**700**	**24**

 # 8-Week Training during summer vacations after 6th semester

8th Semester		Contact Hrs.			Marks			Credits
Code	**Course**	**L**	**T**	**P**	**Int.**	**Ext.**	**Total**	
BITE2-836	Artificial Intelligence and Expert Systems	3	1	0	40	60	100	4
BITE2-837	Artificial Intelligence and Expert Systems Lab.	0	0	4	60	40	100	2
BITE2-838	Project-II	0	0	12	60	40	100	6
Departmental Electives-V (Choose any one)		3	1	*0*	40	60	100	4
BITE2-868	Object Oriented Analysis and Design							
BITE2-869	Big Data							
BITE2-870	Soft Computing							
Total		**6**	**2**	**16**	**200**	**200**	**400**	**16**

Total Credits = 25 + 25 + 23 + 22 + 24 + 22 + 24 + 16 = 181

DATA STRUCTURES

Subject Code: BITE2-302 **L T P C** **Duration: 45 Hrs.**
3 1 0 4

COURSE OBJECTIVES:

To learn the concepts of data structure and algorithms and its implementation. The course has the main ingredients required for a computer science graduate and has all the necessary topics for assessment of data structures and algorithms.

COURSE OUTCOMES:

CO1 Able to comprehend the basic concepts of memory management, data structure, Algorithms and Asymptotic notation.

CO2 Understand and implement linear data structures such as arrays, linked lists, stacks and Queues.

CO3 Understand the concepts of non-linear data structures such as graphs, trees and heaps.

CO4 Able to describe and implement hashing, Searching and Sorting Techniques

UNIT-I (11 Hrs.)

Introduction: Data Structures and data types, Efficient use of memory, Recursion, operations on data structures, time and space complexity of algorithms, Asymptotic Notations.

Arrays: Linear and multi-dimensional arrays and their representation in memory, operations on arrays, sparse matrices and their storage.

UNIT-II (12 Hrs.)

Linked Lists: Singly linked lists, operations on link list, linked stacks and queues, polynomial addition, sparse matrices, doubly linked lists and dynamic storage management, circular linked list,

Stacks and Queues: Concepts of stack and queues, memory representations, operations on stacks and queues, application of stacks such as parenthesis checker, evaluation of postfix expressions, conversion from infix to postfix representation, implementing recursive functions, deque, priority queue, applications of queues. Garbage collection,

UNIT-III (11 Hrs.)

Trees: Basic terminology, sequential and linked representations of trees, traversing a binary tree using recursive and non-recursive procedures, inserting a node, deleting a node, brief introduction to threaded binary trees, AVL trees and B-trees. Representing a heap in memory, operations on heaps, application of heap in implementing priority queue and heap sort algorithm.

Graphs: Basic terminologies, representation of graphs (adjacency matrix, adjacency list), traversal of a graph (breadth-first search and depth-first search), and applications of graphs. Dijkstra's algorithm for shortest path, Minimal Spanning tree.

UNIT-IV (11 Hrs.)

Hashing & Hash Tables: Comparing direct address tables with hash tables, hash functions, concept of collision and its resolution using open addressing and separate chaining, double hashing, rehashing

Searching & Sorting: Searching an element using linear search and binary search techniques, Sorting arrays using bubble sort, selection sort, insertion sort, quick sort, merge sort, heap sort, shell sort and radix sort, complexities of searching & sorting algorithms.

Recommended Books

1. Tenenbaum, Augenstein, & Langsam, 'Data Structures using C and C++', 2nd Edn., Prentice Hall of India, **2009**.
2. Seymour Lipschutz, 'Data Structures, Schaum's Outline Series', 1st Edn., Tata McGraw Hill, **2005**.
3. R.S. Salaria, 'Data Structures & Algorithms Using C++', 3rd Edn., Khanna Book Publishing

Co. (P) Ltd, **2012.**
4. Kruse, 'Data Structures & Program Design', 3rd Edn., Prentice Hall of India, **1994.**
5. Michael T. Goodrich, Roberto Tamassia, & David Mount, 'Data Structures and Algorithms in C++', 2nd Edn., Wiley India, **2016**.
6. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, 'Introduction to Algorithms', 3rd Edn., PHI COURSE Pvt. Ltd-New Delhi, **2009**.
7. Ellis Horowitz, Sartaj Sahni, & Dinesh Mehta, 'Fundamentals of Data Structures in C++', 2nd Edn., Orient Longman, **2008**.
8. Malik, 'Data Structures using C++', 2nd Edn., Cengage COURSE, **2012**.

OBJECT ORIENTED PROGRAMMING USING C++

Subject Code: BITE2-303 **L T P C** **Duration: 36 Hrs.**
3 0 0 3

Course Objectives:
To introduce the principles and paradigms of Object Oriented Programming Language for design and implement the Object Oriented System

Course Outcomes:
CO1 To introduce the basic concepts of object oriented programming language and its representation
CO2 To allocate dynamic memory, access private members of class and the behavior of inheritance and its implementation.
CO3 To introduce polymorphism, interface design and overloading of operator.
CO4 To handle backup system using file, general purpose template and handling of raised exception during programming

UNIT-I
Introduction to C++, C++ Standard Library, Illustrative Simple C++ Programs. Header Files, Namespaces, Application of object oriented programming.
Object Oriented Concepts, Introduction to Objects and Object Oriented Programming, Encapsulation, Polymorphism, Overloading, Inheritance, Abstract Classes, Accessifier (public/ protected/ private), Class Scope and Accessing Class Members, Controlling Access Function, Constant, Class Member, Structure and Class.

UNIT-II
Friend Function and Friend Classes, This Pointer, Dynamic Memory Allocation and Deallocation (New and Delete), Static Class Members, Constructors, parameter Constructors and Copy Constructors, Deconstructors,
Introduction of inheritance, Types of Inheritance, Overriding Base Class Members in a Derived Class, Public, Protected and Private Inheritance, Effect of Constructors and Deconstructors of Base Class in Derived Classes.

UNIT-III
Polymorphism, Pointer to Derived class, Virtual Functions, Pure Virtual Function, Abstract Base Classes, Static and Dynamic Binding, Virtual Deconstructors.
Fundamentals of Operator Overloading, Rules for Operators Overloading, Implementation of Operator Overloading Like <<, >> Unary Operators, Binary Operators.

UNIT-IV
Text Streams and binary stream, Sequential and Random Access File, Stream Input/ Output Classes, Stream Manipulators.
Basics of C++ Exception Handling, Try, Throw, Catch, multiple catch, Re-throwing an Exception, Exception specifications.

Templates: Function Templates, Overloading Template Functions, Class Template, Class Templates and Non- Type Template arguments.

Recommended Books

1. Robert Lafore, 'Object Oriented Programming in Turbo C++', 2nd Edn., The WAITE Group Press, **1994**.
2. Herbert Shield, 'The Complete Reference C ++', 4th Edn., Tata McGraw Hill, **2003**.
3. Shukla, 'Object Oriented Programming in C++', Wiley India, **2008**.
4. H.M. Deitel and P.J. Deitel, 'C++ How to Program', 2nd Edn., Prentice Hall, **1998**.
5. D. Ravichandran, 'Programming with C++', 3rd Ed., Tata McGraw Hill, **2003**.
6. Bjarne Stroustrup, 'The C++ Programming Language', 4th Edn., Addison Wesley, **2013**.
7. R.S. Salaria, 'Mastering Object-Oriented Programming with C++', Salaria Publishing House, **2016**.

DIGITAL CIRCUITS & LOGICAL DESIGN

Subject Code- BITE2-304 **L T P C** **Duration: 36 Hrs.**
3 0 0 3

Course Objectives:

To learn the basic methods for the design of digital circuits and provide the fundamental concepts used in the design of digital systems.

Course Outcomes:

CO1 To represent numerical values and perform number conversions between different number systems. Also acquire knowledge of Boolean algebra and minimization methods for designing combinational Systems.

CO2 Study and analyze the basic logic gates and various logic families. To Analyze and Design digital combinational circuits.

CO3 Analyze and design flip-flops and latches and design sequential systems composed of standard sequential modules, such as counters and registers.

CO4 To acquire Knowledge of the nomenclature and technology in the area of memory devices and about various analog and digital signals with their conversion techniques.

UNIT-I

Number Systems: Binary, Octal, Decimal, Hexadecimal. Number base conversions, 1's, 2's, rth's complements, signed Binary numbers. Binary Arithmetic, Binary codes: Weighted BCD, Gray code, Excess 3 code, ASCII – conversion from one code to another.

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, Quine-McCluskey method - Don't care conditions.

UNIT-II

Logic GATES: AND, OR, NOT, NAND, NOR, Exclusive-OR and Exclusive-NOR. Implementations of Logic Functions using gates, NAND-NOR implementations. Study of logic families like RTL, DTL, DCTL, TTL, MOS, CMOS, ECL and their characteristics.

Combinational Circuits: Design procedure – Adders, Subtractors, Serial adder/ Subtractor, Parallel adder/ Subtractor Carry look ahead adder, BCD adder, Magnitude Comparator, Multiplexer/ Demultiplexer, encoder/decoder, parity checker, code converters. Implementation of combinational logic using MUX.

UNIT-III

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.

UNIT-IV

Memory Devices: Classification of memories, RAM organization, Write operation, Read operation, Memory cycle. Static RAM Cell-Bipolar, RAM cell, MOSFET RAM cell, Dynamic RAM cell. ROM organization, PROM, EPROM, EEPROM, Field Programmable Gate Arrays (FPGA)

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

Recommended Books

1. Thomas L. Floyd, 'Digital Fundamentals', 11th Rev Edn., Pearson Education, Inc, New Delhi, **2014**.
2. Morris Mano, 'Digital Design', Prentice Hall of India Pvt. Ltd, **2001**.
3. Donald P. Leach and Albert Paul Malvino, 'Digital Principles and Applications', 5th Edn., Tata McGraw Hill Publishing Company Limited, New Delhi, **2003**.
4. R.P. Jain, 'Modern Digital Electronics', 3rd Edn., Tata McGraw–Hill Publishing Company Limited, New Delhi, **2003**.
5. Ronald J. Tocci, Neal S. Widmer, Gregory L. Moss, 'Digital System-Principles and Applications', 10th Edn., Pearson Education, **2009.**
6. Subrata Ghosal, 'Digital Electronics', 1st Edn., Cengage COURSE, **2012**.

COMPUTER ARCHITECTURE & ORGANISATION

Subject Code- BITE2-305 **L T P C** **Duration: 36 Hrs.**
3 0 0 3

Course Objectives:

To have a thorough understanding of the basic structure, operation of a digital computer and study the different ways of communicating with I/O devices and standard I/O interfaces, the hierarchical memory system including cache memories and virtual memory.

Course Outcomes:

CO1 Ability to understand how computer hardware has evolved to meet the needs of multiprocessing systems, Instruction Set Architecture: Instruction format, types, various addressing modes, the basic components and design of the CPU: the ALU and control unit.

CO2 Understand the memory organization: SRAM, DRAM, concepts on cache memory, Memory Interleaving, Associative memory, Virtual memory organization.

CO3 Ability to understand the parallelism both in terms of a single processor and multiple processors.

CO4 Understand the I/O Organization: Basics of I/O, Memory-mapped I/O & I/O mapped I/O, types of I/O transfer: Program controlled I/O, Interrupt-driven I/O, DMA.

UNIT-I (11 Hrs.)

General System Architecture: Store program control concept, Flynn's classification of computers (SISD, MISD, MIMD); Multilevel viewpoint of a machine: digital logic, micro architecture, ISA, operating systems, high level language; structured organization; CPU, caches, main memory, secondary memory units & I/O; Performance metrics; MIPS, MFLOPS.

Instruction Set Architecture: Instruction set based classification of processors (RISC, CISC, and their comparison); addressing modes: register, immediate, direct, indirect, indexed; Operations in the instruction set; Arithmetic and Logical, Data Transfer, Machine Control Flow.

UNIT-II (12 Hrs.)

Basic Non Pipelined CPU Architecture: CPU Architecture types (accumulator, register, stack, memory/ register) detailed data path of a typical register based CPU, Fetch-Decode-Execute cycle (typically 3 to 5 stage); microinstruction sequencing, implementation of control unit, Enhancing performance with pipelining. Hardwired control design method, Micro programmed control unit.

UNIT-III (11 Hrs.)

Memory Hierarchy & I/O Techniques: The need for a memory hierarchy (Locality of reference principle, Memory hierarchy in practice: Cache, main memory and secondary memory, Memory parameters: access/ cycle time, cost per bit); Main memory (Semiconductor RAM & ROM organization, memory expansion, Static & dynamic memory types); Cache memory (Associative & direct mapped cache organizations. Allocation & replacement polices, segments, pages & file organization, virtual memory.

UNIT-IV (11 Hrs.)

Introduction to Parallelism: Goals of parallelism (Exploitation of concurrency, throughput enhancement); Amdahl's law; Instruction level parallelism (pipelining, super scaling –basic features); Processor level parallelism (Multiprocessor systems overview).

Computer Organization [80x86]: Instruction codes, computer register, computer instructions, timing and control, instruction cycle, type of instructions, memory reference, register reference. I/O reference, Basics of Logic Design, accumulator logic, Control memory, address sequencing, micro-instruction formats, micro-program sequencer, Stack Organization, Instruction Formats, Types of interrupts; Memory Hierarchy. Programmed I/O, DMA & Interrupts.

Recommended Books

1. David A. Patterson and John L. Hennessy, 'Computer Organization and Design', 2nd Edn., Morgan Kauffmann Publishers, **1997**.
2. John P. Hayes, 'Computer Architecture and Organization', 3rd Edn., TMH, **1998**.
3. William Stallings, 'Operating Systems Internals and Design Principles', 4th Edn., Prentice-Hall Upper Saddle River, New Jersey, **2001**.
4. Carl Hamacher and Zvonko Vranesic, 'Computer Organization', 5th Edn., SafwatZaky, **2002**.
5. A.S. Tanenbaum, 'Structured Computer Organisation', 4th Edn., Prentice-Hall of India, Eastern Economic Edition, **1999**.
6. W. Stallings, 'Computer Organisation & Architecture: Designing for Performance', 4th Edn., Prentice-Hall International Edition, **1996**.
7. M. Mano, 'Computer Architecture & Organisation', Prentice-Hall, **1990**.
8. Nicholas Carter, 'Computer Architecture', T.M.H., **2002**.

DISCRETE STRUCTURES

Subject Code- BITE2-306 **L T P C** **Duration: 45 Hrs.**
3 1 0 4

Course Objectives:

To learn the ability to distinguish between the tractability and intractability of a given computational problem. To be able to devise fast and practical algorithms for real-life problems using the algorithm design techniques and principles learned in this course.

Course Outcomes:

CO1 To study various fundamental concepts of Set Theory and Logics.
CO2 To study the Functions and Combinatorics.
CO3 To study and understand the Relations, diagraphs and

CO4 To study the Algebraic Structures.

UNIT-I (11 Hrs.)

Sets, Relations and Functions: Introduction, Combination of Sets, ordered pairs, proofs of general identities of sets, relations, operations on relations, properties of relations and functions, Hashing Functions, equivalence relations, compatibility relations, partial order relations.

Basic Logic: Propositional logic, Logical connectives, Truth tables, Normal forms (conjunctive and disjunctive), Validity of well-formed formula, Propositional inference rules (concepts of modus ponens and modus tollens), Predicate logic, Universal and existential quantification, Limitations of propositional and predicate logic.

UNIT-II (10 Hrs.)

Combinatorial Mathematics: Basic counting principles Permutations and combinations Inclusion and Exclusion Principle Recurrence relations, Generating Function, Application.

UNIT-III (12 Hrs.)

Probability Distributions: Probability, Bayes theorem, Discrete & Continuous probability distributions, Moment generating function, Probability generating function, Properties and applications of Binomial, Poisson and normal distributions.

Graph Theory: Graph- Directed and undirected, Eulerian chains and cycles, Hamiltonian chains and cycles Trees, Chromatic number Connectivity, Graph coloring, Plane and connected graphs, Isomorphism and Homomorphism. Applications.

UNIT-IV (12 Hrs.)

Monoids and Groups: Groups Semigroups and monoids Cyclic semigraphs and submonoids, Subgroups and Cosets. Congruence relations on semigroups. Morphisms. Normal subgroups. Dihedral groups.

Rings and Boolean Algebra: Rings, Subrings, morphism of rings ideals and quotient rings. Euclidean domains Integral domains and fields Boolean Algebra direct product morphisms Boolean sub-algebra Boolean Rings Application of Boolean algebra (Logic Implications, Logic Gates, Karnaugh map)

Recommended Books

1. Lipschutz, 'Discrete Mathematics (Schaum Series)', 3rd Edn., McGraw Hill, **2009**.
2. Alan Doerr and Kenneth Levarseur, 'Applied Discrete Structures for Computer Science', Galgotia Publications, **2009**.
3. N. Ch SN Iyengar, V.M. Chandrasekaran, 'Discrete Mathematics', 1st Edn., Vikas Publication House, **2003**.
4. S. Santha, 'Discrete Mathematics and Graph Theory', 1st Edn., Cengage COURSE.
5. Kenneth H. Rosen, 'Discrete Mathematics and its Applications', 7th Edn., McGraw Hill, **2008**.
6. C.L. Liu, 'Elements of Discrete Mathematics', 4th Edn., McGraw Hill, **2012**.
7. Satinder Bal Gupta, 'Discrete Mathematics and Structures', 4th Edn., Laxmi Publications, **2008**.

DATA STRUCTURES LAB.

Subject Code: BITE2-307 **L T P C**
0 0 2 1

COURSE OUTCOMES:

CO1 To introduce the basic concepts of Data structure, basic data types, searching and sorting based on array data types.

CO2 To introduce the structured data types like Stacks and Queue and its basic operation's implementation

CO3 To introduces dynamic implementation of linked list
CO4 To introduce the concepts of Tree and graph and implementation of traversal algorithms.

PRACTICALS

1. Write a program for Linear search methods.
2. Write a program for Binary search methods.
3. Write a program for insertion sort, selection sort and bubble sort.
4. Write a program to implement Stack and its operation.
5. Write a program for quick sort.
6. Write a program for merge sort.
7. Write a program to implement Queue and its operation.
8. Write a program to implement Circular Queue and its operation.
9. Write a program to implement singly linked list for the following operations: Create, Display, searching, traversing and deletion.
10. Write a program to implement doubly linked list for the following operations: Create, Display, inserting, counting, searching, traversing and deletion.
11. Write a program to implement circular linked list for the following operations: Create, Display, inserting, counting, searching, traversing and deletion.
12. Write a program to implement insertion, deletion and traversing in B tree

OBJECT ORIENTED PROGRAMMING USING C++ LAB.

Subject Code- BITE2-308 **L T P C**
0 0 2 1

PRACTICALS

1. Classes and Objects- Write a program that uses a class where the member functions are defined inside a class.
2. Classes and Objects- Write a program that uses a class where the member functions are defined outside a class.
3. Classes and Objects- Write a program to demonstrate the use of static data members.
4. Classes and Objects- Write a program to demonstrate the use of const data members.
5. Constructors and Destructors- Write a program to demonstrate the use of zero argument and parameterized constructors.
6. Constructors and Destructors- Write a program to demonstrate the use of dynamic constructor.
7. Constructors and Destructors- Write a program to demonstrate the use of explicit constructor.
8. Initializer Lists- Write a program to demonstrate the use of initializer list.
9. Operator Overloading- Write a program to demonstrate the overloading of increment and decrement operators.
10. Operator Overloading- Write a program to demonstrate the overloading of binary arithmetic operators.
11. Operator Overloading- Write a program to demonstrate the overloading of memory management operators.
12. Typecasting- Write a program to demonstrate the typecasting of basic type to class type.
13. Typecasting- Write a program to demonstrate the typecasting of class type to basic type.
14. Typecasting- Write a program to demonstrate the typecasting of class type to class type.
15. Inheritance- Write a program to demonstrate the multilevel inheritance

DIGITAL CIRCUIT & LOGICAL DESIGN LAB.

Subject Code- BITE2-309 **L T P C**
0 0 2 1

Course Outcomes:
CO1 To Familiarization with Digital Trainer Kit and associated equipment.
CO2 To Study and design of TTL gates
CO3 To learn the formal procedures for the analysis and design of combinational circuits.
CO4 To learn the formal procedures for the analysis and design of sequential circuits

PRACTICALS

Implementation all experiments with help of Bread- Board.

1. Study of Logic Gates: Truth-table verification of OR, AND, NOT, XOR, NAND and NOR gates; Realization of OR, AND, NOT and XOR functions using universal gates.
2. Half Adder / Full Adder: Realization using basic and XOR gates. 13 13 Punjab Technical University B. Tech. Computer Science Engineering (CSE)
3. Half Subtractor / Full Subtractor: Realization using NAND gates.
4. 4-Bit Binary-to-Gray & Gray-to-Binary Code Converter: Realization using XOR gates.
5. 4-Bit and 8-Bit Comparator: Implementation using IC7485 magnitude comparator chips.
6. Multiplexer: Truth-table verification and realization of Half adder and Full adder using IC74153 chip.
7. Demultiplexer: Truth-table verification and realization of Half subtractor and Full subtractor using IC74139 chip.
8. Flip Flops: Truth-table verification of JK Master Slave FF, T-type and D-type FF using IC7476 chip.
9. Asynchronous Counter: Realization of 4-bit up counter and Mod-N counter using IC7490 & IC7493 chip.
10. Synchronous Counter: Realization of 4-bit up/down counter and Mod-N counter using IC74192 & IC74193 chip.
11. Shift Register: Study of shift right, SIPO, SISO, PIPO, PISO & Shift left operations using IC7495 chip.
12. DAC Operation: Study of 8-bit DAC (IC 08/0800 chip), obtain staircase waveform using IC7493 chip.
13. ADC Operations: Study of 8-bit ADC.

SOFT SKILLS-I

Subject Code: **BHUM0-F91** **L T P C**
0 0 2 1

Course Objectives:
The course aims to cause a basic awareness about the significance of soft skills in professional and interpersonal communications and facilitate an all-round development of personality.

Course Outcomes:
At the end of the course, the student will be able to develop his/her personal traits and expose their personality effectively.

UNIT-1

SOFT SKILLS: Introduction to Soft Skills, Aspects of Soft Skills, Identifying your Soft Skills, Negotiation skills, Importance of Soft Skills, Concept of effective communication.
SELF-DISCOVERY: Self-Assessment, Process, Identifying strengths and limitations, SWOT Analysis Grid.

UNIT-2

FORMING VALUES: Values and Attitudes, Importance of Values, Self-Discipline, Personal Values - Cultural Values-Social Values-some examples, Recognition of one's own limits and deficiencies.

UNIT-3

ART OF LISTENING: Proxemics, Haptics: The Language of Touch, Meta Communication, Listening Skills, Types of Listening, Listening tips.

UNIT-4

ETIQUETTE AND MANNERS: ETIQUETTE- Introduction, Modern Etiquette, Benefits of Etiquette, Taboo topics, Do's and Don'ts for Men and Women. MANNERS- Introduction, Importance of manners at various occasions, Professional manners, Mobile manners. CORPORATE GROOMING TIPS- Dressing for Office: Do's and Don'ts for Men and Women, Annoying Office Habits.

Recommended Books

1. K. Alex, S. Chand Publishers.
2. Butterfield, Jeff, 'Soft Skills for Everyone', Cengage Learning, New Delhi, **2010**.
3. G.S. Chauhan and Sangeeta Sharma, 'Soft Skills', Wiley, New Delhi, **2016**.
4. Klaus, Peggy, Jane Rohman & Molly Hamaker, 'The Hard Truth About Soft Skills', Harper Collins E-books, London, **2007**.
5. S.J. Petes, Francis, 'Soft Skills and Professional Communication', Tata McGraw Hill Education, New Delhi, **2011**.

OPERATING SYSTEMS

Subject Code: BITE2-411 **L T P C** **Duration: 38 Hrs.**
3 0 0 3

Course Objectives:

To understand the services and design of Operating Systems. To understand the organization of file systems and process scheduling and memory management

Course Outcomes:

CO1 Understanding operating system functions, Role of operating system, different structures and views of Operating system.

CO2 Process management CPU scheduling, Scheduling Algorithms, PCB, Process synchronization, Deadlocks, Prevention, Detection and Recovery.

CO3 Memory Management Overlays, Memory management policies, Fragmentation and its types, Portioned memory managements, Paging, Segmentation, Ned of Virtual memories, Page replacement Algorithms, Concept of Thrashing.

CO4 Device Management, I/O system and secondary storage structure, Device management policies, Role of I/O traffic controller File Management File System Architecture, Layered Architecture, Physical and Logical File Systems, Protection and Security. Brief study to multiprocessor and distributed operating systems.

UNIT-I

Introductory Concepts: Operating System functions and characteristics, historical evolution of operating systems, Real time systems, Distributed systems, Methodologies for implementation of O/S service, system calls, system programs, interrupt mechanisms.

Processes: Processes model, process states, process hierarchies, implementation of processes, data structures used such as process table, PCB creation of processes, context switching, exit of processes. Interprocess communication: Race conditions, critical sections, problems of mutual exclusion, Peterson's solution, producer-consumer problem, semaphores, counters, monitors, message passing.

UNIT-II

Process Scheduling: Objective, preemptive vs non- preemptive scheduling, comparative assessment of different algorithms such as round robin, priority bases scheduling, FCFS, SJF, multiple queues with feedback.

Deadlocks: Conditions, modeling, detection and recovery, deadlock avoidance, deadlock prevention.

Memory Management: Multiprogramming with fixed partition, variable partitions, virtual partitions, virtual memory, paging, demand paging design and implementation issues in paging such as page tables, inverted page tables, page replacement algorithms, page fault handling, working set model, local vs global allocation, page size, segmentation and paging.

UNIT-III

File Systems: File type, attributes, access and security, file operations, directory structures, path names, directory operations, implementation of file systems, implementation of file and file operations calls, implementation of directories, sharing of files, disk space management, block allocation, free space management, logical file system, physical file system.

Device Management: Techniques for device management, dedicated devices, shred devices, virtual devices, device characteristics -hardware considerations: input and output devices, storage devices, independent device operation, buffering, multiple paths, device allocation considerations.

UNIT-IV

Distributed Systems: Introduction to II/W and S/W concepts in distributed systems, Network operating systems and NFS, NFS architecture and protocol, client- server model, distributed file systems, RPC- Basic operations, parameter passing, RPC semantics in presence of failures threads and thread packages.

Case Studies: LINUX / UNIX Operating System and Windows based operating systems. Recent trends in operating system

Recommended Books

1. J.L. Peterson & Silberschatz, 'Operating System Concepts', 4th Edn., Addison Wesley, **1994**.
2. Brinch, Hansen, 'Operating System Principles', PHI, **2001**.
3. A.S. Tenanbaum, 'Operating System', PHI.
4. Dhamdhere, 'Systems Programming & Operating Systems', Tata McGraw-Hill Education, **1999**.
5. Gary Nutt, 'Operating Systems Concepts', 3rd Edn., Pearson/Addison Wesley, **2004**.
6. William Stallings, 'Operating System', 5th Edn., Pearson Education India, **2005**.

DATABASE MANAGEMENT SYSTEMS-I

Subject Code- BITE2-412 **L T P C** **Duration: 45 Hrs.**
3 0 0 3

Course Objectives:

To familiarize the students with Data Base Management system

Course Outcomes:

CO1 To provide introduction to database systems and various models.
CO2 To provide introduction to relational model and SQL
CO3 To understand about Query Processing and Transaction Processing.
CO4 To learn the concept of failure recovery and concurrency control

UNIT-I (11 Hrs.)

Introduction to Database Systems: File Systems Versus a DBMS, Advantages of a DBMS, Describing and Storing Data in a DBMS, Database System Architecture, DBMS Layers, Data independence.

Data Models: Relational Model, Network Model, Hierarchical Model, ER Model: Entities, Attributes and Entity Sets, Relationships and Relationship Sets, Constraints, Weak Entities, Class Hierarchies, Aggregation, Conceptual Database Design with the ER Model, Comparison of Models.

UNIT-II (12 Hrs.)

The Relational Model: Introduction to the Relational Model, ER to Relational Model Conversion, Integrity Constraints over Relations, Enforcing Integrity Constraints, Relational Algebra, Relational Calculus, Querying Relational Data

Relational Query Languages: SQL: Basic SQL Query, Creating Table and Views, SQL as DML, DDL and DCL, SQL Algebraic Operations, Nested Queries, Aggregate Operations, Integrity Constraints in SQL.

UNIT-III (11 Hrs.)

Database Design: Functional Dependencies, Reasoning about Functional Dependencies, Normal Forms, Schema Refinement, 1NF, 2NF, 3NF, BCNF, 4NF, 5NF, Domain Key Normal Forms.

Transaction and Concurrency Management: ACID Properties, Serializability, Two-phase Commit Protocol, 2PL protocol, Lost Update Problem, Inconsistent Read Problem. Concurrency Control, Lock Management, Read-Write Locks, Deadlocks Handling.

UNIT-IV (11 Hrs.)

Physical Data Organization: File Organization and Indexing, Index Data Structures, Hashing, B-trees, Clustered Index, Sparse Index, Dense Index, Fixed length and Variable Length Records.

Database Protection: Threats, Access Control Mechanisms: Discretionary Access Control, Mandatory Access Control, Grant and Revoke, Role Based Security, Encryption and Digital Signatures.

Recommended Books

1. Abraham Silberschatz, Henry F. Korth, S. Sudarshan, 'Database System Concepts', 6th Edn., Tata McGraw Hill, **2011**.
2. Ramez Elmasri, ShamkantNavathe, 'Fundamentals of Database Systems', 5th Edn., Pearson Education, **2010**.
3. C.J. Date, 'An Introduction to Database Systems', Pearson Education, 8th Edn., **2006**.
4. Alexis Leon, Mathews Leon, 'Database Management Systems', Leon Press, 1st Edn., **2008.**
5. S.K. Singh, 'Database Systems Concepts, Design and Applications', 2nd Edn., Pearson Education, **2011**.
6. Raghu Ramakrishnan, Johannes Gehrke, 'Database Management Systems', 3rd Edn., Tata McGraw Hill, **2014.**

COMPUTER NETWORKS-I

Subject Code- BITE2-413 **L T P C** **Duration: 38 Hrs.**
3 0 0 3

Course Objectives:

This course introduces students to computer networks and concentrates on building a firm foundation for understanding Data Communications and Computer Networks. It is based around the OSI Reference Model which deals with the major issues in the bottom four

(Physical, Data Link, Network and Transport) layers of the model. They are also introduced to the areas of Network Security and Mobile Communications.

Course Outcomes:

CO1 to provide knowledge about various types of networking, networks and network topologies. Also acquire knowledge about concepts of OSI reference model and real world protocol suite such as TCP/IP.

CO2 Outline the basic network configurations, various Multiplexing and Switching Techniques.

CO3 Analyse, specify and design the Addressing Schemes and routing strategies for an IP based networking infrastructure

CO4 Operations of TCP/UDP, FTP, HTTP, SMTP, SNMP and Security and protection issues etc.

UNIT-1

Introduction to Computer Networks: Data Communication System and its components, Data Flow, Computer network and its goals, Types of computer networks: LAN, MAN, WAN, Wireless and wired networks, broadcast and point to point networks, Network topologies, Network software: concept of layers, protocols, interfaces and services, ISO-OSI reference model, TCP/IP reference model.

UNIT-II

Physical Layer: Concept of Analog & Digital Signal, Bandwidth, Transmission Impairments: Attenuation, Distortion, Noise, Data rate limits: Nyquist formula, Shannon Formula, multiplexing: Frequency Division, Time Division, Wavelength Division, Introduction to Transmission Media: Twisted pair, Coaxial cable, Fiber optics, Wireless transmission (radio, microwave, infrared), Switching: Circuit Switching, Message Switching, Packet Switching & their comparisons.

Data Link Layer: Framing, Error detection and correction codes: checksum, CRC, hamming code, Data link protocols for noisy and noiseless channels, Sliding Window Protocols: Stop & Wait ARQ, Go-Back-N ARQ, Selective repeat ARQ, Data link protocols: HDLC and PPP.

UNIT-III

Medium Access Sub-Layer: Static and dynamic channel allocation, Random Access: ALOHA, CSMA protocols, Controlled Access: Polling, Token Passing, IEEE 802.3 frame format, Ethernet cabling, Manchester Encoding, collision detection in 802.3, Binary exponential back off algorithm.

Network Layer: Design issues, IPv4 classful and classless addressing, subnetting, IPv6, Routing algorithms: distance vector and link state routing, Congestion control: Principles of Congestion Control, Congestion prevention policies, Leaky bucket and token bucket algorithms

UNIT-IV

Transport Layer: Elements of transport protocols: addressing, connection establishment and release, flow control and buffering, multiplexing and de-multiplexing, crash recovery, introduction to TCP/UDP protocols and their comparison, Sockets.

Application Layer: World Wide Web (WWW), Domain Name System (DNS), E-mail, File Transfer Protocol (FTP), SMTP, POP, HTTP, Introduction to Network security

Recommended Books

1. Andrew S. Tanenbaum, 'Computer Networks', 4th Edn., Pearson Education, **2002**.
2. Behrouz A. Forouzan, 'Data Communication & Networking', 4th Edn., Tata McGraw Hill, **2006**.
3. James F. Kurose and Keith W. Ross, 'Computer Networking', 3rd Edn., Pearson Education, **2012**.
4. W. Stallings, 'Data & Computer Communications', 9th Edn., PHI, **2014**.

5. Douglas E. Comer, 'Internetworking with TCP/IP', Volume-I, 2nd Edn., Prentice Hall, India, **1996**.
6. Greg Tomsho, 'Guide to Networking Essentials', 6th Edn.., Cengage COURSE, **2011**.
7. Michael W. Graves, 'Handbook of Networking', Cengage COURSE.

DESIGN & ANALYSIS OF ALGORITHMS

Subject Code- BITE2-414 **L T P C** **Duration: 45 Hrs.**
3 1 0 4

Course Objectives:
To learn the ability to distinguish between the tractability and intractability of a given computational problem. To be able to devise fast and practical algorithms for real-life problems using the algorithm design techniques and principles learned in this course.

Course Outcomes:
CO1 Basic ability to analyze algorithms and to determine algorithm correctness and time efficiency class.
CO2 Ability to apply and implement learned algorithm design techniques and data structures to solve problems.
CO3 Differentiate between various algorithms for sorting, searching, and selection and know the concepts of tractable and intractable problems and the classes P, NP and NP-complete problems.
CO4 Analysis of Geometric algorithms (range searching, convex hulls, segment intersections, closest pairs) Know various Text pattern matching, tries, KMP Algorithm.

UNIT-I (11 Hrs.)

Introduction: Algorithms and its Properties, Time and space complexity of an algorithm. Comparing the performance of different algorithms for the same problem. Different orders of growth. Asymptotic notation. Polynomial vs. Exponential running time.
Basic Algorithm Design Techniques. Divide-and-conquer, greedy, Backtracking, Branch and Bound, dynamic programming and randomization. Overall technique with example, problems and algorithms illustrating the use of these techniques.

UNIT-II (12 Hrs.)

Graph Algorithms. Graph traversal: breadth-first search (BFS) and depth-first search (DFS). Applications of BFS and DFS. Topological sort. Shortest paths in graphs: Dijkstra and Bellman-Ford (Single source shortest path, And All pair shortest path (Floyd Warshal algorithm). Minimum spanning Trees: Prim's and Kruskal Algorithm.

UNIT-III (11 Hrs.)

Sorting and Searching. Binary search in an ordered array. Sorting algorithms such as Merge sort, Quick sort, Heap sort, Radix Sort, and Bubble sort with analysis of their running times. Lower bound on sorting, searching and Merging, Median and order statistics.
NP-Completeness. Definition of class P, NP. NP-hard and NP-complete problems. 3SAT is NP-complete. Proving a problem to be NP-complete using polynomial-time reductions. Examples of NP-complete problems. Approximation algorithms for various NP-complete problems: TSP, Hamiltonian Cycle, Knapsack.

UNIT-IV (11 Hrs.)

Advanced Topics. Pattern matching algorithms: Knuth-Morris-Pratt algorithm, Brute Force. Algorithms in Computational Geometry: Convex hulls: Jarvin March and Graham Scan. Integer and polynomial arithmetic. Matrix multiplication: Strassen's algorithm.

Recommended Books

1. J. Kleinberg and E. Tardos, 'Algorithm Design', 1st Edn., Pearson Publications, **2005**.

2. H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, Introduction to Algorithms, 3rd Edn., The MIT Press Ltd, **2009**.
3. S. Dasgupta, C.H. Papadimitriou, and U.V. Vazirani, 'Algorithms', McGraw Hill Education, **2006**.
4. Michael T. Goodrich and Roberto Tamassia, 'Algorithm Design: Foundations, Analysis, and Internet Examples', 1st Edn., Wiley India Pvt Ltd, **2006**.
5. V. Aho, J.E. Hopcroft, and J.D. Ullman, 'The Design and Analysis of Computer Algorithms', 1st Edn., Pearson India, **1974**.
6. Donald Knuth, 'The Art of Computer Programming', Volumes 1, 2 and 3, 2nd Edn., Addison-Wesley Professional, **1998**.

MICROPROCESSORS & ASSEMBLY LANGUAGES

Subject Code- BITE2-415 **L T P C** **Duration: 37 Hrs.**
3 0 0 3

Course Objectives:
The course is intended to give students good understanding of internal architectural details and functioning of microprocessors.

Course Outcomes:
CO1 To study and differentiate microprocessors, microcomputers and microcontrollers.
CO2 To understand the detailed architecture of 8085 and learn assembly language programming using the instruction set of 8085.
CO3 To study the interfacing of microprocessors with memory and I/O devices.
CO4 To give an overview of higher order microprocessors and know about the various applications of microprocessors using the interfaces

UNIT-I

Introduction: Introduction to Microprocessors, Microcomputers, Microcontrollers, history and classification of microprocessors, recent microprocessors.

UNIT-II

Microprocessor Architecture: 8085 microprocessor Architecture. Bus structure, I/O, Memory & Instruction execution sequence & Data Flow, Instruction cycle. System buses, concept of address Bus, Data Bus & Control Bus, Synchronous & Asynchronous buses.

Instruction Set & Assembly Languages Programming: Introduction, instruction & data formats, addressing modes, status flags, 8085 instructions, Data transfer operations, Arithmetic operations, Logical operations, Branch operations.

UNIT-III

I/O and Memory Interfaces: Interfacing of memory chips, memory mapped and isolated I/O structure**,** Data transfer modes: Programmable, interrupt initiated and DMA, Interfacing of I/O devices, Serial & parallel interface, Detail study of 8251 I/O Processor & 8255 programmable peripheral interfaces.

UNIT-IV

Basic Architecture of Higher Order Microprocessors: Basic introduction to 8086 family, pin description and architecture of 8086.

Microprocessor Applications: Interfacing of keyboards and seven segment LED display, Microprocessor controlled temperature system (MCTS), Study of traffic light system, stepper motor controller, differentiate microprocessors, microcomputers and microcontrollers using their applications.

Recommended Books

1. Ramesh Gaonkar, '8085 Microprocessor', 5th Edn., PHI Publications, **2002**.
2. Daniel Tabak, 'Advanced Microprocessors', 2nd Edn., McGraw Hill, Inc., **1995**.

3. Douglas V. Hall, 'Microprocessors and Interfacing: Programming and Hardware', Tata McGraw Hill, **1986**.
4. Charles M. Gilmore, 'Microprocessors: Principles and Applications', McGraw Hill, 2nd Edn., **1995**.
5. Ayala Kenneth, 'The 8086 Microprocessor Programming and Interfacing', 1st Edn., Cengage COURSE, **2007.**

DATABASE MANAGEMENT SYSTEMS-I LAB.

Subject Code- BITE2-416 **L T P C**
0 0 4 2

Course Outcomes:
CO1 To understand basic DDL, DML, DCL commands
CO2 To understand the SQL queries using SQL operators
CO3 To understand the concept of relational algebra, date and group functions
CO4 To learn view, cursors and triggers.

PRACTICALS

1. Write the queries for Data Definition Language (DDL) in RDBMS.
2. Write the queries for Data Manipulation Language (DML) in RDBMS.
3. Write the queries for Data Control Language (DCL) in RDBMS.
4. Write SQL queries using logical operations (=,etc)
5. Write SQL queries using SQL operators
6. Write SQL query using character, number, date and group functions
7. Write SQL queries for relational algebra
8. Write SQL queries for extracting data from more than one table
9. Write SQL queries for sub queries, nested queries
10. Concepts for ROLL BACK, COMMIT & CHECK POINTS
11. Case studies on normalization

COMPUTER NETWORKS-I LAB.

Subject Code- BITE2-417 **L T P C**
0 0 2 1

PRACTICALS

1. Write specifications of latest desktops and laptops.
2. Familiarization with Networking Components and devices: LAN Adapters, Hubs, Switches, Routers etc.
3. Familiarization with Transmission media and Tools: Co-axial cable, UTP Cable, Crimping Tool, Connectors etc.
4. Preparing straight and cross cables.
5. Study of various LAN topologies and their creation using network devices, cables and computers.
6. Configuration of TCP/IP Protocols in Windows and Linux.
7. Implementation of file and printer sharing.
8. Designing and implementing Class A, B, C Networks
9. Subnet planning and its implementation
10. Installation of ftp server and client.

DESIGN & ANALYSIS OF ALGORITHM LAB.

Subject Code- BITE2-417 **L T P C**
0 0 2 1

Course Objectives:
To get a first-hand experience of implementing well-known algorithms in a high-level language. To be able to compare the practical performance of different algorithms for the same problem.

PRACTICALS

1. Code and analyse to compute the greatest common divisor (GCD) of two numbers.
2. Code and analyse to find the median element in an array of integers.
3. Code and analyse to find the majority element in an array of integers.
4. Code and analyse to sort an array of integers using Heap sort.
5. Code and analyse to sort an array of integers using Merge sort.
6. Code and analyse to sort an array of integers using Quick sort.
7. Code and analyse Knapsack problem using dynamic programming
8. Code and analyse to find the shortest path for single source shortest path using dynamic programming.
9. Code and analyse to find the shortest path for All pair shortest path using dynamic programming.
10. Code and analyse to do a depth-first search (DFS) on an undirected graph. Implementing an application of DFS such as to find the topological sort of a directed acyclic graph.
11. Code and analyse to do a breadth-first search (BFS) on an undirected graph. Implementing an application of BFS such as (i) to find connected components of an undirected graph, OR (ii) to check whether a given graph is bipartite.
12. Code and analyse to find the minimum spanning tree in a weighted, undirected graph.
13. Code and analyse to find all occurrences of a pattern P in a given string S using KMP Method
14. Code and analyse to compute the convex hull of a set of points in the plane.

MICROPROCESSORS AND ASSEMBLY LANGUAGES LAB.

Subject Code-BITE2-419 **L T P C**
0 0 2 1

Course Outcomes:
CO1 Understanding different steps to develop program such as Problem definition, Analysis, Design of logic, Coding, Testing, Maintenance
CO2 To be able to apply different logics to solve given problem.
CO3 To be able to write program using different implementations for the same problem
CO4 Use of programming language constructs in program implementation

PRACTICALS

1. Introduction to 8085 kit.
2. Addition of two 8-bit numbers, sum 8-bit.
3. Subtraction of two 8-bit numbers.
4. Find 1's complement of 8-bit number.
5. Find 2's complement of 8-bit number.
6. Shift an 8-bit no. by one bit.
7. Find Largest of two 8-bit numbers.
8. Find Largest among an array of ten numbers (8-bit).
9. Sum of series of 8-bit numbers.

10. Introduction to 8086 kit.
11. Addition of two 16-bit numbers, sum 16-bit.
12. Subtraction of two 16-bit numbers.
13. Find 1's complement of 16-bit number.
14. Find 2's complement of 16-bit number.

SOFT SKILLS-II

Subject Code: **BHUM0-F92** **L T P C**
0 0 2 1

Course Objectives:
The course aims to address various challenges of communication as well as behavioural skills faced by individual at work place and organizations. Also, it aims to enhance the employability of the students.

Course Outcomes:
At the end of the course the student will be able to understand the importance of goal setting. They will also be able to handle stress in their lives and future in a better way.

UNIT-1

DEVELOPING POSITIVE ATTITUDE: Introduction. Formation of attitude. Attitude in workplace. Power of positive attitude. Examples of positive attitudes. Negative attitudes. Examples of negative attitude. overcoming negative attitude and its consequences.
IMPROVING PERCEPTION: Introduction. Understanding perception. perception and its application in organizations.

UNIT-2

CAREER PLANNING: Introduction. Tips for successful career planning. Goal setting-immediate, short term and long term. Strategies to achieve goals. Myths about choosing career.

UNIT-3

ART OF READING: Introduction. Benefits of reading. Tips for effective reading. the SQ3R technique. Different stages of reading. determining reading rate of students. Activities to increase the reading rate. Problems faced. Becoming an effective reader.

UNIT-4

STRESS MANAGEMENT: Introduction. meaning. positive and negative stress. Sources of stress. Case studies. signs of stress. Stress management tips. Teenage stress.

Recommended Books

1. K. Alex, S. Chand Publishers.
2. Rizvi, M. Ashraf, 'Effective Technical Communication', McGraw Hill.
3. Mohan Krishna & Meera Banerji, 'Developing Communication Skills', Macmillan.
4. Kamin, Maxine, 'Soft Skills Revolution: A Guide for Connecting with Compassion for Trainers, Teams & Leaders', Pfeiffer & Amp; Company, Washington, DC, **2013.**

SYSTEM ANALYSIS AND DESIGN

Subject Code: **BITE2-520** **L T P C** **Duration: 45 Hrs.**
3 1 0 4

Course Objectives:
The course has been designed to provide a solid foundation of systems principles and an understanding of how business function, while heightening students to the issues analysts face daily.

UNIT-I

1. Introduction: System definition and concepts: Characteristics and types of automated systems, Manual and Real-life Business sub-systems: Production, Marketing, Personal, Material, Finance Systems models types of models: Systems environment and boundaries, Real- time and distributed systems, Basic principles of successful systems

2. Systems Analyst: Role and need of systems analyst, Qualifications and responsibilities, Systems Analyst as an agent of change.

3. System Development Cycle: Introduction to systems development life cycle (SDLC): Various phases of development: Analysis, Design, Development, Implementation, Maintenance Systems documentation considerations: Principles of systems documentation, Types of documentation and their importance, Enforcing documentation discipline in an organization.

UNIT-II

4. System Planning: Data and fact gathering techniques: Interviews, Group communication, Presentations, Site visits. Feasibility study and its importance Types of feasibility reports, System Selection plan and proposal Prototyping Cost-Benefit and analysis: Tools and techniques

5. Systems Design and Modeling:

Process modeling, Logical and physical design, Design representation, Systems flowcharts and structured charts, Data flow diagrams, Common diagramming conventions and guidelines using DFD and ERD diagrams. Data Modeling and systems analysis, Designing the internals: Program and Process design, Designing Distributed Systems.

UNIT-III

6. Input and Output: Classification of forms: Input/output forms design, User-interface design, Graphical interfaces.

7. Modular and Structured Design: Module specifications, Module coupling and cohesion, Top-down and bottom-up design.

8. System Implementation and Maintenance: Planning considerations, Conversion methods, producers and controls, System acceptance Criteria, System evaluation and performance, Testing and validation, Systems qualify Control and assurance, Maintenance activities and issues.

UNIT-IV

9. System Audit and Security: Computer system as an expensive resource: Data and Strong Media Procedures and norms for utilization of computer equipment, Audit of computer system usage, Audit trails, Types of threats to computer system and control measures: Threat to computer system and control measures, Disaster recovery and contingency planning.

10. Object Oriented Analysis and Design: Introduction to Object Oriented Analysis and design life cycle, object modeling: Class Diagrams, Dynamic diagramming modeling: state diagram, Dynamic modeling: sequence. Case study of the following systems
(I) Inventory Control, (II) Railway Reservation System, (III) University Management System, (IV) Hospital management System.

Recommended Books

1. Elias M. Awad, ‘System Analysis and Design’.
2. Perry Edwards, ‘System Analysis and Design’.
3. Ames A. Senn, ‘Analysis and Design of Information Systems’.

Course Outcomes

Upon successful completion of this course, the student will be able to:

1. Define and describe the five phases of the system development life cycle.
2. State at least five expected benefits from systems projects.
3. Explain at least three ways in which information systems support business requirements.

4. Describe how systems analysts interact with users, management, and other information systems professionals.
5. Develop data flow diagrams and decision tables.
6. Perform a feasibility study.
7. Evaluate systems development alternatives.
8. Solve realistic systems analysis problems.
9. Determine methods for evaluating the effectiveness and efficiency of a system.
10. Work as an effective team member on assigned projects.

PROGRAMMING IN JAVA

Subject Code: **BITE2-521** **L T P C** **Duration: 45 Hrs.**
3 1 0 4

Course Objectives:
This course will provide the knowledge of Java and prepare students to be in a position to write object oriented programs in Java.

UNIT-I

1. Overview of Java: Object oriented programming, two paradigms, abstraction, the three OOP principles, Java class libraries.

2. Date Types, Variables and Arrays:
Integers, floating-point types, characters, Boolean, Iterates, Variable, Data types and casting, automatic type promotion in expressions, arrays.

3. Operators and Control Statements:
Arithmetic operators, bit wise operators, relational operators, Boolean logical operators, the ? Operators, operator precedence, Java's selection statements, iteration statements, jump statements.

UNIT-II

4. Introduction to Classes:
Class fundamentals, declaring object reference variable, Introducing methods, method. Constructors, this keyword, garbage collection, the finalize.

5. Methods and Classes:
Overloading methods, using objects as parameters, recursion.

6. Inheritance:
Inheritance basics, using super, method overriding, dynamic method dispatch, using abstract Classes, using final with inheritance, Package and Interfaces, Package access protection, importing packages.

UNIT-III

7. Exception Handling: Exception handling fundamentals, Exception types, Uncaught Exceptions Using try and catch, multiple catch clauses, nested try statements, throw, finally Java's built-in exceptions. Exceptions, creating your own exception sub classes, using

8. Multithreaded Programming:
The Java thread model, the main thread, creating thread, creating multiple threads, using is alive and join, Thread priorities, synchronization, inter thread communications, suspending resuming and stopping threads.

9. String Handling:
The string constructors, string length, special string operations, character extraction, string comparison, searching string, modifying string, data conversion, changing the case of characters, string buffer.

UNIT-IV

10. I/O and Applets:
I/O Basics, Reading Console Input, Writing Console Output, Reading and Writing Files, Applet Fundamentals, Applet Architecture, The HTML Applet tag, Passing parameters to Applets.

11. Networking:
Networking basics, Java and the Net, TCP/IP Client Sockets URL, URL Connection, TCP/IP Server Sockets, Database connectivity.

Recommended Books

1. Herbert Schildt, 'The Complete Reference Java2', McGraw Hill.
2. Joyce Farrell, 'Java for Beginners', Cengage Learning.
3. Deitel and Deitel, 'Java: How to Program', 6th Edn., Pearson Education.
4. James Edward Keogh, Jim Keogh, 'J2EE: The Complete Reference', McGraw Hill.
5. Khalid A. Mughal, Torill Hamre, Rolf W. Rasmussen, 'Java Actually', Cengage Learning.
6. Shirish Chavan, 'Java for Beginners', 2nd Edn., Shroff Publishers.

Course Outcomes

Upon successful completion of this course, the student will be able to:
1. Knowledge of the structure and model of the Java programming language, (knowledge)
2. Use the Java programming language for various programming technologies (understanding)
3. Develop software in the Java programming language, (application)
4. Evaluate user requirements for software functionality required to decide whether the Java programming language can meet user requirements (analysis)
5. Propose the use of certain technologies by implementing them in the Java programming language to solve the given problem (synthesis)
6. Choose an engineering approach to solving problems, starting from the acquired knowledge of programming and knowledge of operating systems. (Evaluation)

COMPUTER NETWORKS–II

Subject Code: **BITE2-522** **L T P C** **Duration: 45 Hrs.**
3 1 0 4

Course Objectives:
The objective of the course is to offer good understanding of the concepts of network security, wireless, Adhoc and various emerging network technologies. Course Contents:

UNIT-I

1. Network Security: Fundamentals of network security, Basics of IPv6, IPsec: overview of IPsec, IP and IPv6, Authentication header (AH), Encapsulating Security Payload (ESP).
2. Internet Key Exchange (IKE): History, Photuris, Simple Key-management for Internet protocols (SKIP), IKE phases, IKE encoding.

UNIT-II

3. Adhoc Networks: Features, advantages and applications, Adhoc versus Cellular networks, Network architecture, Protocols: MAC protocols, Routing protocols, Technologies.

UNIT-III

4. Wireless Communication Systems: Evolution, examples of wireless communication systems, 2G Cellular networks, Evolution for 2.5G TDMA Standards, IS-95B for 2.5G CDMA.

UNIT-IV

5. 3G Wireless Networks: Wireless local loop (WLL), Local Multipoint Distribution System (LMDS), Wireless local Area Networks (WLANs), Bluetooth and Personal Area Networks.

6. Wireless System Design: Introduction, Frequency reuse, channel assignment strategies, handoff strategies, interference and system capacity, improving coverage and capacity in cellular systems.

Recommended Books

1. Theodore S. Rappaport, 'Wireless Communication: Principles and Practices', 2nd Edn., Pearson Education.
2. Charlie Kaufman, Radio Perlman, Mike Speciner, 'Neywork Security', 2nd Edn., PHI.
3. Sunilkumar S. Manvi, Mahabaleshwar S. Kakkasageri, 'Wireless and Mobile Networks: Concepts and Protocols', Wiley India.
4. Michael A. Gallo & William M. Hancock, 'Computer Communications and Networking Technologies', Cengage Learning / Thomson Brooks / Cole.
5. S. Keshav, 'An Engineering Approach to Computer Networking', Pearson Education. Mayank Dave, 'Computer Networks', Cengage Learning.

Course Outcomes

Upon successful completion of this course, the student will be able to:

1. Able to define the Fundamentals of network security, Characteristics of IPv6 and their addressing format and schemes.
2. Acquire the Knowledge about various concepts of IPSec and able to explain about various concepts of Ad-hoc and Cellular Networks.
3. Acquire the Knowledge about wireless communication systems and their generations with different Technologies.
4. Able to explain about Third Generation Networks, their Technologies, wireless System Design and their various strategies

CYBER LAWS AND IPR

Subject Code: **BITE2-556** **L T P C** **Duration: 45 Hrs.**
3 1 0 4

Course Objectives:

The objective of the course is to offer good understanding of Cyber law and awareness of intellectual property rights.

1. Basics of Computer & Internet Technology: Internet, ISP & domain name; Network Security; Encryption Techniques and Algorithms; Digital Signatures.

2. Introduction to Cyber World: Introduction to Cyberspace and Cyber Law; Different Components of cyber Laws; Cyber Law and Netizens.

3. E-Commerce: Introduction to E-Commerce; Different E-Commerce Models; E-Commerce Trends and Prospects; E-Commerce and Taxation; Legal Aspects of E-Commerce.

4. Intellectual Property Rights: IPR Regime in the Digital Society; Copyright and Patents; International Treaties and Conventions; Business Software Patents; Domain Name Disputes and Resolution.

5. IT ACT 2000: Aims and Objectives; Overview of the Act; Jurisdiction; Role of Certifying Authority; Regulators under IT Act; Cyber Crimes-Offences and Contraventions; Grey Areas of IT.

6. Project Work: Candidates will be required to work on a project. At the end of the course students will make a presentation and submit the project report.

Recommended Books

1. Nandan Kamath, 'A Guide to Cyber Laws & IT Act 2000 with Rules & Notification'.
2. Keith Meril l& Deepti Chopra (IK Inter.), 'Cyber Cops, Cyber Criminals & Internet'.
3. Diane Row Land, 'Information Technology Law'.

4. Vakul Sharma, 'Handbook of Cyber Laws', McMillian.

Course Outcomes:

Upon successful completion of this course, the student will be able to:

1. To describe how can stay safe from cybercrime.
2. To understand the features of Intellectual Property rights.
3. To understand about attacks and their properties.

COMPUTER GRAPHICS

Subject Code: **BITE2-557** **L T P C** **Duration: 45 Hrs.**
3 1 0 4

Course Objectives:

Understanding the fundamental graphical operations and the implementation on computer, get a glimpse of recent advances in computer graphics, understanding user interface issues that make the computer easy for the novice to use.

UNIT-I

1. Introduction: Computer Graphics and its applications, Elements of a Graphics, Graphics Systems: Video Display Devices, Raster Scan Systems, Random Scan Systems, Input devices.

2. Basic Raster Graphics: Scan Conversion-Point plot technique, Line drawing, Circle generating and Ellipse generating algorithms.

UNIT-II

3. Two-dimensional Geometric Transformations: Basic Transformations-Translation, Rotation and Scaling, Matrix Representation and Homogeneous Coordinates, Composite Transformations, Reflection and Shearing transformations.

4. Clipping: Window to viewport transformation, Clipping Operations- Point Clipping, Line Clipping, Polygon Clipping and Text Clipping.

UNIT-III

5. Filling Techniques: Scan line algorithms, Boundary-fill algorithm, Flood-fill algorithm, Edge fill and fence fill algorithms

6. Elementary 3D Graphics: Plane projections and its types, Vanishing points, Specification of a 3D view.

UNIT-IV

7. Visibility: Image and object precision, Hidden edge/surface removal or visible edge/surface determination techniques; z buffer algorithms, Depth sort algorithm, Scan line algorithm and Floating horizon technique.

8. Advance Topics: Introduction of Rendering, Raytracing, Antialiasing, Fractals, Gourard and Phong shading.

Recommended Books

1. Donald Hearn and M. Pauline Baker, 'Computer Graphics', 2nd Edn., PHI/Pearson Education.
2. Zhigandxiang, Roy Plastock, Schaum's Outlines, 'Computer Graphics', 2nd Edn., Tata McGraw Hill.
3. C. Foley, Van Dam, Feiner and Hughes, 'Computer Graphics Principles & Practice', 2nd Edn., Pearson Education.

Course Outcomes:

Upon successful completion of this course, the student will be able to:

1. To provide comprehensive introduction about computer graphics system, design algorithms and two dimensional transformations.

2. To make the students familiar with techniques of clipping, three dimensional graphics and three dimensional transformations.
3. The computer graphics course prepares students for activities involving in design, development and testing of modeling, rendering, shading and animation.

LINUX AND SHELL PROGRAMMING

Subject Code: **BITE2-558** **L T P C** **Duration: 45 Hrs.**
3 1 0 4

Course Objectives:
This Linux Bash Shell Programming training course is designed to give delegates practical experience in developing and writing LINUX shell scripts. Most of the built-in shell commands are introduced together with the main program control structures. This course is not suitable for C shell programmers.

UNIT–I

Linux introduction and file system - Basic Features, Advantages, Installing requirement, Basic Architecture of Unix/Linux system, Kernel, Shell. Linux File System-Boot block, super block, Inode table, data blocks, How Linux access files, storage files, Linux standard directories. Commands for files and directories cd, ls, cp, md, rm, mkdir, rmdir, pwd, file, more, less, creating and viewing files using cat, file comparisons – cmp&comm, View files, disk related commands, checking disk free spaces. Partitioning the Hard drive for Linux, Installing the Linux system, System startup and shut-down process, init and run levels.

UNIT–II

Essential linux commands Understanding shells, Processes in linux-process fundamentals, connecting processes with pipes, tee, redirecting input output, manual help, Background processing, managing multiple processes, changing process priority with nice, scheduling of processes at command, cron, batch commands, kill, ps, who, sleep, Printing commands, find, sort, touch, file, file related commands-ws, sat, cut, dd, etc. Mathematical commands- bc, expr, factor, units. Creating and editing files with vi, joe & vim editor

UNIT–III

System administration Common administrative tasks, identifying administrative files – configuration and log files, Role of system administrator, managing user accounts-adding & deleting users, changing permissions and ownerships, Creating and managing groups, modifying group attributes, Temporary disable user's accounts, creating and mounting file system, checking and monitoring system performance file security & Permissions, becoming super user using su. Getting system information with name, host name, disk partitions & sizes, users, kernel. Backup and restore files, reconfiguration hardware with kudzu, installing and removing packages with rpm command. Configure X-windows desktop-redhat-config-Xfree86, understanding XF86config file, starting & using X desktop. KDE & Gnome graphical interfaces, changing X settings.

UNIT–IV

Shell Programming- Basic of shell programming, Various types of shell available in Linux, comparisons between various shells, shell programming in bash, read command, conditional and looping statements, case statements, parameter passing and arguments, Shell variables, system shell variables, shell keywords, Creating Shell programs for automate system tasks. Simple filter commands – pr, head, tail, cut, paste, sort, uniq, tr. Flter using regular expressions – grep, egrep, and sed. awk programming – report printing with awk.

Recommended Books

1. Sumitabha Das, 'UNIX – Concepts & Applications', 3rd Edn., Tata McGraw Hill.

2. Graham Glass & King Ables, 'Unix for Programmers and Users', 3rd Edn., Pearson Education India. (Low Prices Edition).
3. Cristopher Negus, 'Red Hat Linux 9 Bible', IDG Books India Ltd.

Course Outcomes

1. Upon successful completion of this course, the student will be able to:
2. Writing simple scripts to enhance basic command output
3. Using the various shell quoting mechanisms appropriately
4. Manipulating shell variables and user-defined variables in scripts
5. Implementing conditional execution facilities
6. Using the shell's built-in loop constructs where appropriate
7. Writing scripts to trap user interrupts
8. User defined Functions
9. Developing menu-driven shell scripts

PROGRAMMING IN JAVA LAB.

Subject Code: **BITE2-523** **L T P C**
3 1 0 4

1. Implementation of classes.
2. Implementation of inheritance.
3. Implementation of packages and interfaces.
4. Implementation of threads.
5. Using exception handling mechanisms.
6. Implementation of Applets.
7. Implementation of mouse events, and keyboard events.
8. Implementing basic file reading and writing methods.
9. Using basic networking features.
10. Connecting to Database using JDBC.

COMPUTER NETWORKS–II LAB.

Subject Code: **BITE2-524** **L T P C**
3 1 0 4

1. To configure the IP address for a computer connected to LAN and to configure network parameters of a web browser for the same computer.
2. To plan IPv6 address scheme for a local area network comprising of 'n' terminals.
3. To develop programs for implementing / simulating routing algorithms for Adhoc networks.
4. To install any one open source packet capture software like wireshark etc.
5. To configure Wireless Local Loop.
6. To plan Personal Area Network.
7. To configure WLAN.
8. To configure Adhoc networks.
9. To install and configure wireless access points.

SOFT SKILLS-III

Subject Code: BHUM0-F93 **L T P C** **Duration: 25 Hrs.**
0 0 2 1

UNIT-1

ART OF WRITING: Introduction, Importance of Writing Creative Writing, Writing tips, Drawback of written communication.

ART OF BUSINESS WRITING: Introduction, Business Writing, Business Letter, Format and Styles, Types of business letters, Art of writing correct and precise mails, Understand netiquette.

UNIT-2

BODY LANGUAGE: Introduction- Body Talk, Forms of body language, uses of body language, Body language in understanding Intra and Inter-Personal Relations, Types of body language, Gender differences, Gaining confidence with knowledge of Kinesics.

UNIT-3

TEAM BUILDING AND TEAM WORK: Introduction, Meaning, Characteristics of an effective team, Role of a Team Leader, Role of Team Members, inter group Collaboration Advantages, Difficulties faced, Group Exercises-Team Tasks and Role-Play, Importance of Group Dynamics.

UNIT-4

TIME MANAGEMENT: Introduction, the 80-20 Rule, three secrets of Time Management, Time Management Matrix, Effective Scheduling, Time Wasters, Time Savers, Time Circle Planner, Difficulties in Time Management, Overcoming Procrastination.

Recommended Books

1. K. Alex, S. Chand Publishers.
2. R.C. Sharma and Krishna Mohan, 'Business Correspondence and Report Writing', TMH, New Delhi, **2016**.
3. N. Krishnaswami and T. Sriraman, 'Creative English for Communication', Macmillan.
4. Penrose, M. John, et al., 'Business Communication for Managers', Thomson South Western, New Delhi, **2007**.
5. Holtz, Shel, 'Corporate Conversations', PHI, New Delhi, **2007.**

NETWORK PROGRAMMING

Subject Code: **BITE2-626** **L T P C** **Duration: 45 Hrs.**
3 1 0 4

Course Objectives:

To familiarize students with advanced concepts of networks, network programming in UNIX environment.

UNIT-I

1. OSI model, client server model, TCP/IP protocols, Introduction to Unix; Process, groups, job control and non-job control shells, reliable and unreliable signals, shell Programming.
2. 2.Inter process communication in Unix, pipes, half duplex and full duplex pipes, FIFOs, properties of pipes and FIFOs, POSIX message queues, system V message queues, semaphores, shared memory, mmap function and its use, RPC, authentication, timeout and retransmission, call semantics, XDR.

UNIT-II

3. Communication Protocol: Introduction, TCP, IP, XNS, SNA, NetBIOS, OSI protocols, comparisons.

UNIT-III

4. Introduction to Berkeley sockets, socket addressing, TCP and UDP socket functions, sockets and Unix signals, socket implementation, client and server examples for TCP and UDP and their behavior under abnormal conditions.

UNIT-IV

5. Socket options, IPv4, IPv6, TCP, I/O multiplexing, Unix I/O models, select and poll functions.
6. System V Transport Layer, interface – Introduction Transport End Point address, TLI.

Recommended Books

1. W.R. Stevens, B. Fenner & A.M. Rudoff, 'Unix Network Programming', Vol. I, 3rd Edn., Pearson Education.
2. W.R. Stevens, 'Unix Network Programming', Vol. II, 2nd Edn., Pearson Education.
3. Comer and Stevens, 'Internetworking with TCP/IP', Vol. -I, -II, -III, PHI.
4. Christian Benvenuti, 'Understanding Linux Network Internals', O'Reilly.
5. W.R. Stevens, 'Advanced Programming in Unix Environment', Pearson Education.

Course Outcomes:

Upon successful completion of this course, the student will be able to:

1. To understand the use of client/server architecture, inter process communication and to explain the basic communication protocols.
2. To understand elementary socket system calls, advanced socket system calls and Java Socket API and to explain the basic concepts relating to TCP and UDP based sockets.
3. To understand File transfer protocol, remote login using pseudo terminal and RPC.

SOFTWARE ENGINEERING

Subject Code: **BITE2-627** **L T P C** **3 1 0 4** **Duration: 45 Hrs.**

Course Objectives:

To enable the students to learn the principles and methodologies followed to develop a good software.

UNIT-I

1. Introduction: Evolution and impact of Software engineering, Software crisis, Principles of Software Engineering, Feasibility study Software Life Cycle Models: Waterfall, prototyping, Evolutionary, and Spiral models, Comparison of software models.

UNIT-II

2. Scheduling and Planning: Management Activities, Project planning and control, cost estimation, project scheduling using PERT and GANTT charts. Requirement Analysis: Functional and Non-functional requirements, Requirements gathering, Requirements analysis and specification.

UNIT-III

3. Software Design: Basic principles of software design, modularity, cohesion, coupling and layering, function-oriented software design: DFD and Structure chart, object modeling using UML, Object-oriented software development, Design specifications, Design metrics, Verification and validation, User Interface design.

Coding: Coding standards and Code review techniques, Coding styles, Coding metrics.

Software Testing: Fundamentals of testing, Types of software testing, White-box, and black-box testing, test case design techniques, mutation testing, Testing metrics.

UNIT-IV

4. Reliability: Software reliability metrics, reliability growth modeling. Software Quality Management: Risk Management, Quality management, ISO and SEI CMMI, Six Sigma,

Computer aided software engineering, Software maintenance, Software Configuration Management, Component-based software developments.

Recommended Books

1. Pressman, 'Software Engineering: A Practitioner's Approach', 3rd Edn., TMH, **2004.**
2. Flecher and Hunt, 'Software Engineering and CASE: Bridging and Culture Gap', **2000**.
3. Shepperd, 'Software Engineering, Metrics', Vol.-1 (EN), McMillan, **1999**.
4. Robert S. Arnold, 'Software Re–engineering', IEEE Computer Society, **1994**.
5. Pankaj Jalote, 'An Integrated Approach to Software Engineering', 3rd Edn., Narosa Publishers, **2006**.
6. Ghezzi, Cario, 'Fundamentals of Software Engineering', 2nd Edn., PHI, **2002**.
7. Sommerville, Ian, 'Software Engineering', 7th Edn., Pearson Education, **2004**.
8. Watts Humphrey, 'Managing Software Process', 2nd Edn., Pearson Education, **2003**.
9. James F. Peters and Witold Pedrycz, 'Software Engineering – An Engineering Approach', 1st Edn., Wiley, **2010**.
10. Mouratidis and Giorgini, 'Integrating Security and Software Engineering Advances and Future', IGP.

Course Outcomes:

Upon successful completion of this course, the student will be able to:

1. To study how software engineering principles, evolve and to analyze the various software models that can be followed to develop a software.
2. To understand the software analysis and design step of software development.
3. To study coding, testing and reliability of a software.
4. To highlight the various management activities and related terms of a software.

MOBILE APP DEVELOPMENT

Subject Code: **BITE2-659** **L T P C** **Duration: 45 Hrs.**
3 1 0 4

Course Objectives:

This course teaches students how to build mobile apps for Android, iOS, and Windows 8, the trinity that is today's mobile operating platforms.

UNIT-I

1. Characteristics of Mobile Applications: Architecture and working of Android, iOS and Windows phone8 operating system. User-interface design for mobile applications and managing application data. Integrating cloud services, networking, OS and hardware into mobile-applications. Addressing enterprise requirements in mobile applications: performance, scalability, modifiability, availability and security.

UNIT-II

2. Introduction to Android Development Environment: What Is Android? Advantages and Future of Android, Frameworks, Tools and Android SDK. Installing Java, Android Studio, SDK Manager Components and updating its platforms, AVD Manager, Understanding Java SE and the Dalvik Virtual Machine. The Directory Structure of an Android Project, Common Default Resources Folders, The Values Folder, Leveraging Android XML.

User Interface Widgets: Text controls, Button controls, Toggle buttons, Images. Notification and Toast: Parameters on Intents, Pending intents, Status bar notifications, Toast notifications. Menus & Dialogs: Localization, Options menu, Context menu; Alert dialog, Custom dialog, Dialog as Activity.

Lists: Using string arrays, Creating lists, Custom lists. Location and Maps: Google maps, Using GPS to find current location.

UNIT-III

3. Application Development in Android: App Components (Intents and Intent Filters, activities, services, Content Providers, App Widgets, Processes and Threads), App resources, App Manifest and User interface, Action Bar, Content Sharing, Multi-Platform Designs, Animation and graphics, computation, Media and Camera, Location and sensors, Connectivity, Text and Input, Data Storage, Administration and Web Apps.
Publishing Your App: Preparing for publishing, Signing and preparing the graphics, publishing to the Android Market.

UNIT-IV

4. Introducing SQLite: SQLite Open Helper and creating a database, Opening and closing a database Cursors and its types, Working with cursors Inserts, updates, and deletes.
Database Connectivity: SQLite Data Types, Content Values, Adding, Updating and Deleting Content, Content provider: introduction, Query providers.

Recommended Books

1. Jeffmcwherter, Scott Go Well, 'Professional Mobile Application Development', 1st Edn., Wiley, **2012**.
2. Belen Cruz, Zapata, 'Android Studio Application Development', 2nd Edn., Packt Publishing, **2016**.
3. Reto Meier, 'Professional Android 4 Application Development', Wrox Publication, **2012**.
4. Onur Cinar, 'Beginning Android 4', 1st Edn., Apress Publication, **2012**.
5. David Mark, 'Beginning iPhone Development with Swift', Apress Publication, **2014**.
6. Android Developer Site: http://developer.android.com/index.html.

Course Outcomes:

Upon successful completion of this course, the student will be able to:
1. To be familiar with the Architecture of various Mobile Application Platform.
2. Ability to work on Android using various forms and menus.
3. Knowledge to publish your developed Mobile Application.
4. Using SQLite for connection to database type facilities.

CRYPTOGRAPHY & NETWORK SECURITY

Subject Code: **BITE2-660** **L T P C** **Duration: 45 Hrs.**
3 1 0 4

Course Objectives:

The main objective of this course is to make student able to understand the basic concepts, services, threats and principles in network security, various security services and mechanisms in the network protocol stack.

UNIT-I

1. Security trends, Attacks and services, Classical crypto systems, Different types of ciphers, LFSR sequences, Basic Number theory, Congruence, Chinese Remainder theorem, Modular exponentiation, Fermat and Euler's theorem, Legendre and Jacobi symbols, Finite fields, continued fractions.

UNIT-II

2. Simple DES, Differential crypto analysis, DES –Modes of operation, Triple DES, AES, RC4, RSA, Attacks–Primality test –factoring.

UNIT-III

3. Discrete Logarithms, Computing discrete logs, Diffie-Hellman key exchange, ElGamal Public key cryptosystems, Hash functions, Secure Hash, Birthday attacks, MD5, Digital signatures, RSA, ElGamal DSA.

UNIT-IV

4. Authentication applications–Kerberos, X.509, PKI–Electronic Mail security–PGP, S/MIME–IP security–Web Security–SSL, TLS, SET. Intruders, Malicious software, viruses and related threats, Firewalls, Security Standards.

Recommended Books

1. Wade Trappe, Lawrence C. Washington, 'Introduction to Cryptography with Coding Theory', 2nd Edn., Pearson, **2007**.
2. William Stallings, 'Cryptography and Network Security Principles and Practices', 4th Edn., Pearson/PHI, **2006**.
3. W. Mao, 'Modern Cryptography–Theory and Practice', 2nd Edn., Pearson Education, **2007**.
4. Charles P. Pfleeger, Shari Lawrence Pfleeger, 'Security in Computing', 3rd Edn., Prentice Hall of India, **2006**.
5. Behrouz Forouzan, 'Cryptography & Network Security', 2nd Edn., McGraw Hill, **2011**.

Learning Outcomes:

Upon successful completion of this course, the student will be able to:

1. To understand the tools and description of java scripts
2. To XML and the study of Java beans and introduction to EJB'S
3. To understand Java servelet HTTP package and security issues.
4. To understand JSP Application Development and database programming using JDBC.

WEB TECHNOLOGIES

Subject Code: **BITE2-661** **L T P C** **3 1 0 4** **Duration: 45 Hrs.**

Course Objectives:

This course covers the major web development technologies and techniques. The topics covered include HTML5, Cascading Style Sheets, client side programming using JavaScript, HTML5, server side programming using ASP.NET, constructing web databases using PHP, XML, web services.

1. Internet and World Wide Web: Introduction, Internet Addressing, ISP, types of Internet Connections, Introduction to WWW, WEB Browsers, WEB Servers, URLs, HTTP, WEB Applications, Tools for web site creation.

2. HTML5: Introduction to HTML5, Lists, adding graphics to HTML5 page, creating tables, linking documents, forms, frames, Cascading Style sheets.

3. Java Script: Introduction, programming constructs: variables, operators and expressions, conditional checking, functions and dialog boxes, JavaScript DOM, creating forms, introduction to Cookies, Jquery.

4. AJAX: Introduction, HTTP Request, XMLHttpRequest, AJAX Server Script.

5. PHP: Introduction, syntax, statements, operators, PHP and MySQL, PHP and AJAX.

6. Introduction to ASP.net, J2EE, POJO, Java servlets and JSP.

Recommended Books

1. Deitel, Nieto, Lin and Sadhu, 'XML How to Program?', Pearson Education.
2. Ivan Bayross, 'Web Enabled Commercial Application Development using HTML, DHTML, JavaScript, Perl CGI, BPB'.
3. Steven M. Schafer, 'HTML, CSS, JavaScript, Perl, Python and PHP', Wiley India Textbooks.
4. Paul S. Wang, G. Keller, S. Katila, 'An Introduction to Web Design + Programming', Cengage Learning.
5. Jeffery C. Jackson, 'Web Technologies: A Computer Science Perspective', Pearson Education.

6. Stephen Walhter, Kevin Hoffman, Nate Dudek, 'ASP.NET 4 Unleashed', Pearson Education.
7. James Keogh, 'ASP.NET 2.0 Demystified', McGraw Hill.
8. Scott Mitchell, Sams, 'Teach Yourself ASP.NET 4 in 24 Hours', SAMS, Pearson Education.
9. Robin Nixon, 'Learning PHP, MySQL, and JavaScript', Shroff/O'Reilly.

Course Outcomes:

Upon successful completion of this course, the student will be able to:

1. Analyze a web page and identify its elements and attributes.
2. Create web pages using XHTML and Cascading Styles sheets.
3. Build dynamic web pages using JavaScript (client side programming).
4. Write non-trivial programs using C#.
5. Build interactive web applications using ASP.NET and C#.
6. Build web applications using PHP.
7. Construct and manipulate web databases using ADO.NET.
8. Create XML documents.
9. Create XML Schema.
10. Build and consume web services.

CLOUD COMPUTING

Subject Code: **BITE2-662** **L T P C** **Duration: 45 Hrs.**
3 1 0 4

Course Objectives:

This course gives an introduction to cloud computing and its techniques, issues, and its services that will lead to design and development of a simple cloud service.

Introduction: Cloud Computing in a Nutshell, Layers and Types of Clouds, Desired Formats of Cloud, Cloud Infrastructure Management, Challenges and Risks

Virtualization: Virtualization of Computing, Storage and Resources.

Cloud Services: Introduction to Cloud Services IaaS, PaaS and SaaS

Software as a Service (SaaS): Evolution of SaaS, Challenges of SaaS Paradigm, SaaS Integration Services, SaaS Integration of Products and Platforms, Business – to Business Integration B2Bi Services.

Infrastructure as a Services (IaaS): Introduction, Background & Related Work, Virtual Machines Provisioning and Manageability, Virtual Machine Migration Services, VM Provisioning and Migration in Action, Provisioning in a Cloud Context.

Platform as a service (PaaS): Integration of Private and Public Cloud, Technologies and Tools for Cloud Computing, Resource Provisioning Services.

Map Reduce Programming models and Implementations: Introduction, Map Reduce Programming Model, Major Map Reduce Implementations for the Cloud, Map Reduce Impacts

Migrating into a Cloud: Cloud Services for Individuals, Cloud Services Aimed at the Mid-Market, Enterprise Class Cloud Offering, Introduction to File System & Hadoop.

Management and Monitoring: Accounts Monitoring, User profiles in Cloud, Resource Allocation and Pricing in Cloud.

Security: Introduction, Cloud Storage: from LANs to WANs, Technologies for Data Security in Cloud Computing, Security Concerns, Legal issues and Aspects, Securing the Private and Public Cloud Architecture.

Recommended Books

1. Rajkumar Buyya, James Broberg, Andrzej M. Goscinski, 'Cloud Computing: Principles and Paradigms', Wiley Publications.
2. Toby Velte, Anthony Velte, 'Cloud Computing: A Practical Approach', McGraw Hill Osborne Media.
3. George Reese, 'Cloud Application Architectures: Building Applications and Infrastructure in the Cloud', O'Reilly Publication.
4. John Rhoton, 'Cloud Computing Explained: Implementation Handbook for Enterprises', Recursive Press.

Course Outcomes:

Upon successful completion of this course, the student will be able to:

1. Understand the hardware, software concepts and architecture of cloud computing.
2. Contrast the key technical and commercial issues concerning cloud computing versus traditional software models.
3. Realize the importance of virtualization technology in support of cloud computing.
4. Explore the issues related to cloud computing.

ENTERPRISE RESOURCE PLANNING

Subject Code: **BITE2-663** **L T P C** **Duration: 45 Hrs.**
3 1 0 4

Course Objectives:

To learn the concepts of Enterprise resource Planning. The course has all the required contents that are necessary for a graduate to understand the different strategies of an organization.

1. ERP AND TECHNOLOGY: Introduction, Related Technologies, Business Intelligence, E-Commerce and E-Business, Business Process Reengineering, Data Warehousing, Data Mining, OLAP, Product life Cycle management, SCM, CRM.

2. ERP IMPLEMENTATION: Implementation Challenges, Strategies, Life Cycle, Methodologies Package selection, Project Teams, Vendors and Consultants, Data Migration, Project management.

3. ERP IN ACTION & BUSINESS MODULES: Operation and Maintenance, Business Modules, Finance, Manufacturing, Human Resources, Plant maintenance, Materials Management, Quality management, Marketing, Sales, Distribution and service.

4. ERP Application: Enterprise Application Integration, ERP II, Total quality management

ERP CASE STUDY: SAP AG, JD Edwards.

Recommended Books

1. Alexis Leon, 'ERP Demystified', 2nd Edn., Tata McGraw Hill, **2008**.
2. Mary Sumner, 'Enterprise Resource Planning', Pearson Education, **2007**.
3. Jim Mazzullo, 'SAP R/3 for Everyone', 2nd Edn., Pearson, **2007**.
4. Jose Antonio Fernandz, 'The SAP R /3 Handbook', Tata McGraw Hill, **2000**.
5. Biao Fu, 'SAP BW: A Step-by-Step Guide', 1st Edn., Pearson Education, **2003**.

Course Outcomes:

Upon successful completion of this course, the student will be able to:

1. To understand the concepts of ERP and its related technologies.
2. To understand the implementation of ERP in an organization.
3. To have a deep understanding of different business modules of an organization.
4. To have a basic understanding of applications of ERP and various ERP software's.

PARALLEL AND DISTRIBUTED COMPUTING

Subject Code: **BITE2-664** **L T P C** **Duration: 45 Hrs.**
3 1 0 4

Course Objectives:
To learn the advanced concepts of Parallel and Distributed Computing and its implementation for assessment of understanding the course by the students

UNIT-I

1. Introduction: Parallel Processing, Parallel Computers, Shared Memory Multiprocessing, Distributed Parallel Computing, Message Passing, Parallelism, Speedup, Utilizing Temporal Parallelism Utilizing Data Parallelism, Comparison of Temporal and Data Parallel Processing, Data Parallel Processing with Specialized processor.
2. Processes & Shared Memory Programming: Processes, Shared Memory Programming, forking for Creating Processes, Joining for Process Termination, Process Model Under UNIX.

UNIT-II

3. Basic Parallel Programming Techniques: Loop Splitting, self-scheduling, Contention and Mutual Exclusion, Introduction of Semaphores and Spin –Lock Implementation, Indirect scheduling, Barriers and Race Conditions Overcoming forward and backward data dependencies.
4. Thread-Based Implementation: Thread Management, Mutual Exclusion with Threads, Events and Condition Variables, Deviation Computation with Threads – POSIX/Java Threads.

UNIT-III

5. Distributed Computing Systems: Design goals, Transparencies, Fundamental Issues.
6. Distributed Coordination: Temporal ordering of events and global state detection, Process synchronization and Distributed mutual exclusion algorithms, inter process communication: Message passing model, Remote procedure call, Point to point and Group communication.

UNIT-IV

7. Issues in Distributed systems: Load balancing and Load sharing, Deadlock, MPI (Message passing Interface) and PVM (Parallel Virtual Machine) architecture and features.

Recommended Books

1. Steven Brawer, ‘Introduction to Parallel Programming’.
2. M. Sasikumar, Dinesh Shikhare and P. Ravi Prakash, ‘Introduction to Parallel Processing’.
3. Randy Chow, T. Johnson, ‘Distributed Operating Systems and Algorithms’, Addison Wesley.
4. A.S. Tanenbaum, ‘Distributed Operating Systems’, Prentice Hall.

Course Outcomes:
Upon successful completion of this course, the student will be able to:
1. Understand the concepts and issues related to parallel and distributed system.
2. Understand intricacies of parallel and distributed programming.
3. Design and develop the programs for parallel and distributed environment.
4. Manage security, performance reliability and other issues while designing in parallel and distributed environment.

NETWORK PROGRAMMING LAB.

Subject Code: **BITE2-628** **L T P C**
0 0 2 1

1. To study and implement various network commands like telnet, ftp, etc.

2. To study various system calls.
3. Programs related to interprocess communication.
4. Programs related to message queues.
5. 5.Programs related to pipes.
6. Programs related to file handling.
7. Programs related to process control.
8. Programs using Socket Programming.

SOFTWARE ENGINEERING LAB.

Subject Code: **BITE2-629** **L T P C**
0 0 2 1

1. Study and usage of OpenProj or similar software to draft a project plan.
2. Study and usage of OpenProj or similar software to track the progress of a project.
3. Preparation of a software requirement specification document, design document and testing phases related document for the same problem.
4. Preparation of software configuration management and risk management related documents.
5. Study and usage of any design phase CASE tool.
6. To perform unit testing and integration testing.
7. To perform various white box and black box testing techniques.
8. Testing a website.

SOFT SKILLS-IV

Subject Code: BHUM0-F94 **L T P C** **Duration: 25 Hrs.**
0 0 2 1

UNIT-1

ART OF SPEAKING: Introduction. Communication process. Importance of communication, channels of communication. Formal and informal communication. Barriers to communication. Tips for effective communication. tips for conversation. Presentation skills. Effective multi-media presentation skills. Speeches and debates. Combating nervousness. Patterns and methods of presentation. Oral presentation, planning and preparation.

UNIT-2

GROUP DISCUSSION: Introduction. Importance of GD. Characters tested in a GD. Tips on GD. Essential elements of GD. Traits tested in a GD .GD etiquette. Initiating a GD. Nonverbal communication in GD. Movement and gestures to be avoided in a GD. Some topics for GD.

UNIT-3

PREPARING CV/RESUME: Introduction – meaning – difference among bio-data, CV and resume. CV writing tips. Do's and don'ts of resume preparation. Vocabulary for resume, common resume mistakes, cover letters, tips for writing cover letters.

UNIT-4

INTERVIEW SKILLS: Introduction. Types of interview. Types of question asked. Reasons for rejections. Post-interview etiquette. Telephonic interview. Dress code at interview. Mistakes during interview. Tips to crack on interview. Contextual questions in interview skills. Emotional crack an interview. Emotional intelligence and critical thinking during interview process.

Recommended Books

1. K. Alex, S. Chand Publishers.

2. Lucas, Stephen E., 'The Art of Public Speaking', 11th Edn., International Edn., McGraw Hill Book Co., **2014**.
3. Goleman, Daniel, 'Working with Emotional Intelligence', Banton Books, London, **1998**.
4. Thrope, Edgar and Showick Trope, 'Winning at Interviews', Pearson Education, **2004**.
5. Turk, Christopher, 'Effective Speaking', South Asia Division: Taylor & Francis, **1985**.

BUILDING ENTERPRISE APPLICATIONS

Subject Code: BITE2-730 **L T P C** **Duration: 45 Hrs.**
3 1 0 4

COURSE OBJECTIVES:

This course aims to cover key concepts and design principles related to enterprise application architecture and enterprise application integration. It first provides students a business strategic perspective on adopting enterprise architecture, and then includes topics like layering structure, business logic organization, patterns for object/relational access layers, model-view-control patterns for Web, message-based enterprise application integration, and recent advances in enterprise application architecture.

Unit-I

Introduction to enterprise applications and their types, software engineering methodologies, life cycle of raising an enterprise application, introduction to skills required to build an enterprise application, key determinants of successful enterprise applications, and measuring the success of enterprise applications

Unit-II

Inception of enterprise applications, enterprise analysis, business modeling, requirements elicitation, use case modeling, prototyping, non-functional requirements, requirements validation, planning and estimation

Unit-III

Concept of architecture, views and viewpoints, enterprise architecture, logical architecture, technical architecture - design, different technical layers, best practices, data architecture and design – relational, XML, and other structured data representations, Infrastructure architecture and design elements - Networking, Internetworking, and Communication Protocols, IT Hardware and Software, Middleware, Policies for Infrastructure Management, Deployment Strategy, Documentation of application architecture and design

Unit-IV

Construction readiness of enterprise applications - defining a construction plan, defining a package structure, setting up a configuration management plan, setting up a development environment, introduction to the concept of Software Construction Maps, construction of technical solutions layers, methodologies of code review, static code analysis, build and testing, dynamic code analysis – code profiling and code coverage

Unit-V

Types and methods of testing an enterprise application, testing levels and approaches, testing environments, integration testing, performance testing, penetration testing, usability testing, globalization testing and interface testing, user acceptance testing, rolling out an enterprise application.

Suggested Readings/Books

Anubhav Pradhan, Satheesha B. Nanjappa et. al., 'Raising Enterprise Applications', Wiley India.

LEARNING OUTCOMES:

Upon successful completion of this course, the student will be able to:

1. explains the principles behind different enterprise patterns and apply them for enterprise application development.
2. Describe the key design related issues and principles in enterprise application architecture
3. Explain how different design patterns for enterprise application architecture work.
4. Explain how different design patterns for enterprise application architecture work
5. Explain how the message-based integration pattern can be applied to enterprise application integration

SIMULATION AND MODELING

Subject Code: BITE2-731 **L T P C** **Duration: 45 Hrs.**
3 1 0 4

COURSE OBJECTIVES:

The objectives of this course are to:

1. Introduce students to the simulation and modelling techniques;
2. Provide students with opportunities to develop basic simulation and modeling skills with respect to carrying out research projects using any simulation method on the computer.

Unit-I Introduction

When simulation is appropriate and when not, advantages and disadvantages of simulation, application areas in communication, computer and software design, systems and systems environment, components of a system, discrete and continuous systems, model of a system, types of models, discrete-event simulation, steps in a simulation study. Simulation Examples-Simulation of queueing systems, on-demand and inventory systems, simulation for reliability analysis etc

Unit-II: General Principles

Concepts in discrete event simulation: event scheduling/time advance algorithms, world views. List Processing: properties and operations, data structures and dynamic allocation, techniques

Unit-III: Simulation Software

Integrated environments. Examples and review of some existing software popular and useful in the industry, e.g., Arena, AutoMod, Extend, Flexsim, Micro Saint, ProModel, Quest, SIMUL8, WITNESS etc. Simulation using languages and environments like C++/Java/GPSS/SSF etc. Experimentation and Statistical-Analysis Tools: common features and relevant current products.

Unit-IV: Statistical Models in Simulation

Terms and concepts. Statistical Models. Review of discrete and continuous distributions. Review of Poisson (stationary and non-stationary) processes. Empirical Distributions; Elementary Queueing Theory- Basic Structure of Queueing Models. Input Source (Calling Population). Queue, Queue Discipline, Service Mechanisms. Notations and relationships between L, W, Lq, and Wq. Little's Formula. Role of Exponential Distribution and Properties. Birth and Death Processes. M/M/s queues. Finite queue variation in M/M/s/K models with different s values. Finite Calling Population cases. Queueing Models involving Non-Exponential Distributions: M/G/1, M/D/s, M/Ek/s (involving Erlang distribution), Models without a Poisson Input, Models involving hyper-exponential distributions, Priority Discipline Queueing Models: Preemptive and Non- Preemptive with results, properties and server number variations, Queueing Networks: Equivalence Property. Infinite Queues in Series and Product Form Solutions. Jackson Networks

Unit-V: Application of Queueing Models

Review of Characteristics (calling population system capacity, arrival processes, behavior and disciplines, service times and mechanisms etc.) and notations, Application of Long-Run **Measures of Performance:** Time average in system, average time spent per customer, Little's

Formula and server utilization, costs. Steady State behaviour of Infinite (M/G/1, M/M/c/infinity, M/M/c/N/infinity) and finite (M/M/c/K/K) Calling Population Models, Use of Network of Queues

Unit-VI: Random Number Generation

Properties. Generation of Pseudo-Random Numbers, Techniques for Generation of Pseudo-Random Numbers: Linear Congruential, Combined Linear Congruential, Random Number Streams. Tests for Random Numbers: Frequency Tests and Tests for Autocorrelation. Random Variate Generation- Inverse Transform Techniques for Exponential, Uniform, Weibull, Triangular and for Empirical Continuous Distributions. Acceptance-Rejection Techniques for Poisson (Stationary and Non- Stationary) Distribution and Gamma Distribution. Special Properties like the Direct Transformation for the Normal and Lognormal Distributions, Convolution Method and others.

Unit-VII: Input Modeling

Data collection, Identifying the Distribution with Data: Histograms, Selection of the Appropriate Family of Distributions, Quantile-Quantile Plots.100 Parameter Estimation: Sample Mean and Sample ariance and various biased and unbiased Estimators. Goodness of Fit Tests applied to Simulation inputs: Chi-Square and Chi-Square with Equal Probabilities, Kolmogorov-Smirnov Tests, p- Values and Best Fits. Verification and Validation of Simulation Models- Verification and Validation of Simulation Models. Calibration and Validation: Face Validity, Validation of Assumptions, Input-Out Transformation Validation.

Unit-VIII: Output Analysis of a Single Model

Output analysis and types of simulation. Stochastic Nature of the Output Data. Measures of Performance and Estimation: Point Estimation and Confidence-Interval Estimation. Output Analysis for Terminating Simulations and Estimation of Probabilities. Output Analysis of Steady State Simulations: Initialization Bias, Error Estimation, Replications, Sample Size and Batch Means for Interval Estimation.

Unit-IX: Comparison and Evaluation of Alternative System Designs

Comparison of Two System Designs.; Sampling with Equal and Unequal Variances. Common Random Numbers. Confidence Intervals with Specified Precision. Comparison of Several System Designs: Bonferroni Approaches to Multiple Comparisons and to Screening and to Selection of the Best. MetamodelingL Sample Linear Regression, Testing for Significance, Multiple Linear Regression. Random Number Assignment for Regression. Optimization via Simulation: Robust Heuristics.

Unit-X: Simulation of Computer Systems

Simulation Tools: Process Orientation and Event Orientation. Model Input: Modulated Poisson Process and Virtual-Memory Referencing. High-Level Simulation. CPU and Memory Simulations. Simulation of Computer Networks- Traffic Modeling, Media Access Control: Token- Passing Protocols and Ethernet, Data Link Layer, TCP, Model Construction.

Simulation Languages: Basic Introduction to Special Simulation Languages: -GPSS/ MATLAB/ Network Simulators.

Suggested Readings/Books

1. Jerry Banks, John S. Carson II, Barry L. Nelson and David M. Nicol, 'Discrete-Event System and Simulation', Prentice Hall of India, New Delhi, **2005.**
2. Averill M. Law, 'Simulation Modelling and Analysis (SIE)', Tata McGraw Hill India, **2007.**
3. David Cloud, Larry Rainey, 'Applied Modeling and Simulation', Tata McGraw Hill, India.
4. Gabriel A. Wainer, 'Discrete-event Modelling and Simulation: A Practitioner's Approach', CRC Press, **2009**.

5. Bernard P. Zeigler, Herbert Praehofer, Tag Gon Kim, 'Theory of Modelling and Simulation: Integrating Discrete event and Continuous Complex Dynamic Systems, Academic Press, **2000**.
6. Walter J. Karplus, George A. Bekey, Boris Yakob Kogan, 'Modelling and Simulation: Theory and Practice', Springer, **2003**.
7. Stanislaw Raczynski, 'Modelling and Simulation: The Computer Science of Illusion', Wiley, **2006**.
8. Mohammad Salameh Obaidat, Georgios I. Papadimitriou, 'Applied System Simulation: Methodologies and Application', Springer, **2003**.
9. van Dijk, Nico M., Boucherie, Richard J. (Eds.). 'Queueing Networks: A Fundamental Approach', 798 p. 148 illus. Springer, **2011**.
10. Bhat, U. Narayan, 'An Introduction to Queueing Theory: Modelling and Analysis in Applications', Springer, **2008** (Birkhäuser Boston).
11. James J. Nutaro, 'Building Software for Simulation: Theory and Algorithms, with Applications in C++'. Wiley, **2010**.

LEARNING OUTCOMES:

Upon successful completion of this course, the student will be able to:

Problem formulation-

1. System definition
2. Model translation
3. Verification, validation
4. Experimental design
5. Analysis (Skills)-
   a) use the simulation software to:
   b) carry out simulation tasks;
   c) use graphs to present their results;
   d) write scripting languages to generate other reports.

SIMULATION AND MODELING LAB.

Subject Code: BITE2-732 **L T P C**
0 0 4 2

1. Programming in MATLAB: Introduction, Branching statements, loops, functions, additional data types, plots, arrays, inputs/outputs etc.
2. Introduction regarding usage of any Network Simulator.
3. Practical Implementation of Queuing Models using C/C++.

BUILDING ENTERPRISE APPLICATIONS LAB.

Subject Code: BITE2-733 **L T P C**
0 0 4 2

Case Study: Students should work on one case study related to any enterprise (for Ex: - manufacturing. IT industry. etc.) to implement (and learn to use the tools to accomplish this task) the following (illustrative only)

Note: At the end of finishing each milestone of the case study, the student will write a report mentioning the conclusion drawn from that section, recommendations /or the improvement and any missing links in the enterprise software.

1. Development of Software Requirement Specification (SRS) document to -
   a) Understand or analyze a given business scenario and document the use case diagrams for the given scenario (use tools like MOOSE. smartdraw. visio etc.)

b) Identify the functional and non-functional requirements for the given scenario and document it in the given template

2. Establishment of architecture of enterprise which
   a) Identifies different modules required in the given enterprise application
   b) Create a logical architecture of databases for the given business scenario documented in the use case diagrams
3. Determination of UI tool that best suits given applications
4. Determination of infrastructure (networking. internetworking. software, hardware, middleware, etc.) required for the given scenario.
5. Development of a comprehensive plan for the enterprise application (refer to the Unit IV of BTIT-701)
6. Create test cases (subset) as per the given template
7. Perform the manual and automated testing on the any code base using any testing tools (such as rational rose. QTP etc.)

INFORMATION SECURITY AND RISK MANAGEMENT

Subject Code: BITE2-765 **L T P C** **Duration: 45 Hrs.**
3 1 0 4

COURSE OBJECTIVES:

The objectives of this course are to:

Provide you with a strategic and in-depth knowledge of the issues involved in the emerging field of Information Systems Risk Management. You will learn how to document the risks and threats that are encountered in modern information security applications and to identify the relationship between these risks and the more commonly occurring risks associated with business and project management.

Unit-I

Essentials of Information Security, Security Threats – Intruders, Viruses, Worms, and other Threats, Vulnerabilities, Cyber Crime and Hacker, Security Assessment, Analysis and Assurance, Role of Cryptography, The Data Encryption Standard (DES), Analyzing and Strengthening of DES, Introduction to Advance Encryption Standard (AES).

Unit-II

Concept and Characteristics of Public Key Encryption system, Introduction to Merkle-Hellman Knapsacks, Rivets – Shamir-Adlman (RSA) Encryption. Hash Algorithms, Message Digest Algorithms such as MD4 and MD5, Secure Hash Algorithms such as SH1 and SHA2. Digital Signature Standard.

Unit-III

Network Security Issues such as Impersonation, Message Confidentiality, Message Integrity, Code Integrity, Denial of Service, Firewalls, DMZs, Virtual Private Networks. Web Security, Email Security, Pretty Good Privacy

Unit-IV

Risk Management and Security planning –Risk management Process Overview and Life Cycle Activities, Information Security Life Cycle, Risk Analysis, Cost Benefit Analysis, Risk Assessment Process, Methodology, Threat assessment, Modes of risk analysis – Effective Risk analysis, Risk Mitigation, Qualitative Risk Analysis, Value Analysis, Case Study of IT Organization

Suggested Readings/Books

1. Principles of Cryptography, William Stallings, Pearson Education.
2. Cryptography & Network Security, Atul Kahate, TMH.
3. Matt Bishop, " Computer Security Art and Science", Pearson/PHI, 2002.

4. Thomas R Peltier, Information Security Risk Analysis, CRC Press,2001.

LEARNING OUTCOMES:

Upon successful completion of this course, the student will be able to:

1. Identify and analyses information security threats and risks arising in the Information Systems of diverse industries.
2. Explore and evaluate possible solutions to these risk scenarios acknowledging cost, complexity of implementation and system user impact.
3. Correlate identified information security risks to continuity management issues.
4. Utilize the principles of group dynamics and the people-centered nature of Information Security.
5. Discriminate between commenting on and copying information.
6. Identify and articulate (both in written form and orally to a peer audience) effective Information Systems risk management strategies.

DIGITAL IMAGE PROCESSING

Subject Code: BITE2-766 **L T P C** **Duration: 45 Hrs.**
3 1 0 4

COURSE OBJECTIVES:

The objectives of this course are to:

1. To study the image fundamentals and mathematical transforms necessary for image processing.
2. To study the image enhancement techniques
3. To study image restoration procedures.
4. To study the image compression procedures.

Unit-I: Introduction to Image Processing

Digital Image representation, Sampling & Quantization, Steps in image Processing, Image acquisition, color image representation

Unit-II: Image Transformation & Filtering

Intensity transform functions, histogram processing, Spatial filtering, Fourier transforms and its properties, frequency domain filters, color models, Pseudo coloring, color transforms, Basics of Wavelet Transforms

Unit-III: Image Restoration

Image degradation and restoration process, Noise Models, Noise Filters, degradation function, Inverse Filtering, Homomorphic Filtering

Unit-IV: Image Compression

Coding redundancy, Interpixel redundancy, Psychovisual redundancy, Huffman Coding, Arithmetic coding, Lossy compression techniques, JPEG Compression.

Unit-V: Image Segmentation & Representation

Point, Line and Edge Detection, Thresholding, Edge and Boundary linking, Hough transforms, Region Based Segmentation, Boundary representation, Boundary Descriptors, Regional

Suggested Readings/Books

1. Gonzalez and Woods, 'Digital Image Processing', ISDN 0-201-600-781, Addison Wesley **1992**.
2. Boyle and Thomas, 'Computer Vision, A First Gurse', 2nd Edn, ISBN 0-632-028-67X, Blackwell Science, **1995**.
3. Pakhera Malay K., 'Digital Image Processing and Pattern Recognition', PHI.
4. Trucco & Verri, 'Introductory Techniques for 3-D Computer Vision', Prentice Hall.
5. Low, 'Introductory Computer Vision and Image Processing', McGraw Hill, **1991**, ISBN 0-07-707403-3.

6. Jain, Kasturi and Schunk, 'Machine Vision', McGraw Hill, **1995** ISBN 0070320187.
7. Sonka, Hlavac, Boyle, 'Digital Image Processing and Computer Vision', First Edn., ISBN 978813150557, Cengage Learning, **2011**.

LEARNING OUTCOMES:

Upon successful completion of this course, the student will be able to:

1. Review the fundamental concepts of a digital image processing system.
2. Analyse images in the frequency domain using various transforms.
3. Evaluate the techniques for image enhancement and image restoration.
4. Categorize various compression techniques.
5. Interpret Image compression standards.
6. Interpret image segmentation and representation techniques.

SOFTWARE PROJECT MANAGEMENT

Subject Code: BITE2-767 **L T P C** **Duration: 45 Hrs.**
3 1 0 4

COURSE OBJECTIVES:

The objectives of this course are to:

Software development is a complex process involving such activities as domain analysis, requirements specification, communication with the customers and end users, designing and producing different artifacts, adopting new paradigms and technologies, evaluating and testing software products, installing and maintaining the application at the end-user's site, providing customer support, organizing end-user's training, envisioning potential upgrades and negotiating about them with the customers, and many more. The proposed subject will take students through the various processes involved in project management.

Pre-requisite Knowledge:

The basic understanding of concepts of Software engineering, computer networks and Database concepts.

Unit-I

Project Management Fundamentals: Basic Definitions, Project Stakeholders and Organizational, Influences on Project Management, Project Management Processes, Project Initiating Processes

Unit-II

Planning and Resourcing a Project: Identifying Requirements, Creating the Work Breakdown structure, Developing the Project Schedule, developing a Project Cost Estimate, Planning Quality, Organizing the Project Team, Planning for Potential Risks

Unit-III

Executing and Managing a Project: Project Executing Processes- Acquiring and Developing the Project Team, Managing the Project Team, Managing Stakeholder Expectations, Directing and Managing the Project while assuring Quality

Unit-IV

Project Monitoring and Controlling Processes: Verifying and Controlling Scope, Managing Schedule and Cost, Controlling Quality, Monitoring and Controlling Risks.

Unit-V

Integrated Change Control, Project Closing Process - Closing a Project.

Suggested Readings/Books

1. Software Engineering - Somerville (Addison Wesley).
2. Software Engineering-Pressmen.

Suggested Tools – Rational Team Concert, MS Project

LEARNING OUTCOMES:

Upon successful completion of this course, the student will be able to:

1. Problem Solving and Critical Thinking (PS&CT),
2. Communication and Interpersonal Skills (C&IS),
3. Ethical and Professional Responsibilities (E&PR).

ARTIFICIAL INTELLIGENCE AND EXPERT SYSTEMS

Subject Code: BITE2-836 **L T P C** **Duration: 45 Hrs.**
3 1 0 4

COURSE OBJECTIVES:

The objectives of this course are to:

The objective of this course is to provide the student with an overview of topics in the field of artificial intelligence (AI). The course also provides the student with a working knowledge of designing an expert system and applying expert system technology in designing and analyzing engineering systems.

Unit-I: Introduction

What is intelligence? Foundations of artificial intelligence (AI). History of AI; Problem Solving- Formulating problems, problem types, states and operators, state space, search strategies.

Unit-II: Informed Search Strategies

Best first search, A* algorithm, heuristic functions, Iterative deepening A*(IDA), small memory A*(SMA); Game playing - Perfect decision game, imperfect decision game, evaluation function, alpha-beta pruning

Unit-III: Reasoning

Representation, Inference, Propositional Logic, predicate logic (first order logic), logical reasoning, forward chaining, backward chaining; AI languages and tools - Lisp, Prolog, CLIPS

Unit-IV: Planning

Basic representation of plans, partial order planning, planning in the blocks world, hierarchical planning, conditional planning, representation of resource constraints, measures, temporal constraints

Unit-V: Uncertainty

Basic probability, Bayes rule, Belief networks, Default reasoning, Fuzzy sets and fuzzy logic; Decision making- Utility theory, utility functions, Decision theoretic expert systems.

Unit 6: Inductive learning - decision trees, rule based learning, current-best-hypothesis search, least- commitment search, neural networks, reinforcement learning, genetic algorithms; Other learning methods - neural networks, reinforcement learning, genetic algorithms.

Unit-VII: Communication

Communication among agents, natural language processing, formal grammar, parsing, grammar.

Suggested / Readings & Books

1. Stuart Russell and Peter Norvig, 'Artificial Intelligence – A Modern Approach', Pearson Education Press, **2001**.
2. Kevin Knight, Elaine Rich, B. Nair, 'Artificial Intelligence', McGraw Hill, **2008**.
3. George F. Luger, 'Artificial Intelligence', Pearson Education, **2001**.
4. Nils J. Nilsson, 'Artificial Intelligence: A New Synthesis', Morgan Kauffman, **2002**.

ARTIFICIAL INTELLIGENCE AND EXPERT SYSTEMS LAB.

Subject Code: BITE2-837 **L T P C**
0 0 4 2

1. Write A Program for DEPTH FIRST SEARCH
2. Write A Program for Best First Search
3. Write A Program to Generate the output for A* Algorithm.
4. Write a Lisp Program to solve Water Jug Problem Using Heuristic Function.
5. Write a Program to Show the Tic Tac Toe Game for 0 and X.
6. Write A Program for Expert System by Using Forward Chaining.
7. Write a program to implement tower of Hanoi.
8. Write a program to implement a heuristic search procedure.
9. Write a program to implement a production system.
10. Write a program to implement search problems of 3 x 3 puzzle.

OBJECT ORIENTED ANALYSIS AND DESIGN

Subject Code: BITE2-868 **L T P C** **Duration: 45 Hrs.**
3 1 0 4

COURSE OBJECTIVES:

The objectives of this course are to:

The aim of the course is to teach methods and techniques for analysis and design of information systems based on an object-oriented approach. The course will, furthermore, discuss a variety of perspectives of development of information systems in order explore the relationship between programming, data bases and software engineering.

Unit-I

Introduction to object oriented systems, Classes, Objects, Abstraction, Inheritance, Polymorphism, Encapsulation, Message Sending, Association, Aggregation, Iterative development and the Unified Process (UP), UP phases: Inception, Elaboration, Construction and Transition, Object-oriented metrics

Unit-II

Introduction to UML, Use Cases and functional requirements, Identifying and writing Use Cases, Decomposition of use cases, Modeling System Workflows using Activity Diagrams, modeling a System's Logical Structure using Classes and Class Diagrams, Modeling Interactions using Sequence Diagrams and Communication Diagrams, Timing Diagrams, Interaction Overview Diagrams, Component Diagram, Package diagram, State Machine Diagrams, Deployment Diagrams.

Unit-III

Introduction to Patterns, GoF Patterns, Creational Patterns, Structural Patterns, Behavioral Patterns, Software Architectural patterns, The Observer Pattern, The Template Method Pattern, Factory Patterns: Factory Method and Abstract Factory, The Singleton Pattern, The Iterator Pattern, The Composite Pattern, The Facade Pattern, The State and Strategy patterns, Command Pattern, The Adapter Pattern, The Proxy Pattern, The Decorator Pattern, The Visitor Pattern, AntiPatterns, Patterns for Assigning Responsibilities: GRASP Patterns.

Unit-IV

Domain modeling, assigning responsibility using sequence diagrams, mapping design to code, CASE tools, Unit, Cluster, and System-level testingof Object-oriented programs, Aspect-oriented and Service-oriented software.

Suggested Readings/Books

1. GradyBooch, JamesRumbaugh, Ivar Jacobson, 'The Unified Modelling Language User Guide', Pearson Education.
2. Hans-Erik Eriksson, Magnus Penker, Brian Lyons, David Fado, 'UML 2 Toolkit', WILEY-Dreamtech India Pvt. Ltd.
3. Meilir Page-Jones, 'Fundamentals of Object Oriented Design in UML', Pearson Education.
4. Pascal Roques, 'Modelling Software Systems Using UML2', WILEY- Dreamtech India Pvt. Ltd.
5. Atul Kahate, 'Object Oriented Analysis & Design', McGraw Hill Companies.
6. John W. Satzinger, Robert B. Jackson and Stephen D. Burd, 'Object-Oriented Analysis and Design with the Unified Process', Cengage Learning.
7. Gamma, et. al., 'Design Patterns - Elements of Reusable Object-Oriented Software', Addison-Wesley, **1994.**
8. Craig Larman, 'Applying UML and Patterns: An Introduction to Object-oriented Analysis and Design and Iterative Development', Pearson Education, **1998.**

LEARNING OUTCOMES:

Upon successful completion of this course, the student will be able to:

1. use an object-oriented method for analysis and design
2. analyses information systems in real-world settings and to conduct methods such as interviews and observations
3. have a general understanding of a variety of approaches and perspectives of systems development, and to evaluate other IS development methods and techniques
4. know techniques aimed to achieve the objective and expected results of a systems development process
5. know different types of prototyping
6. know how to use UML for notation

BIG DATA

Subject Code: BITE2-869 **L T P C** **Duration: 45 Hrs.**
3 1 0 4

COURSE OBJECTIVES:

The objectives of this course are to:

1. Store, manage, and analyze unstructured data
2. Select the correct big data stores for disparate data sets
3. Process large data sets using Hadoop to extract value
4. Query large data sets in near real time with Pig and Hive
5. Plan and implement a big data strategy for your organization

Unit-I

Analysis of data at Rest- Hadoop analytics: Limitations of existing distributing systems, Hadoop Approach, Hadoop Architecture, distributed file system: HDFS and GPFS, Internals of Hadoop MR engine, Need for High level language- JAQL and PIG

Unit-II

Introduction to Text Analytics: Using Regular expressions, Using AQL, Sentiment analysis

Unit-III

No SQL: JSON store, MongoDB, RDF, HBASE

Unit-IV

Analytics: Clustering, Classification, Segmentation, Linear regression, ML.

Unit-V

Search: Indexing and Indexing Techniques, create inverted index using JAQL, Lab using Data Explorer.

Unit-VI

Bundling Hadoop job: Application, Use BI tooling to create application, Publish applications.

Unit-VII

Analysis of data in motion – Real time analytics.

Unit-VIII

Introduction to streams computing, Challenges/limitations of conventional Systems, solving a real time analytics problem using conventional system, Challenges to be solved - scalability, thread pooling, etc., Understanding the challenges in handling streaming data from the real world and how to address those using stream computing, Benefits of stream computing in Big Data world, Real-time Analytics Platform (RTAP).

Suggested Readings/Books

1. Chris Eaton, Paul Zikopoulos, 'Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data'.
2. Frank J. Ohlhorst, 'Big Data Analytics: Turning Big Data into Big Money'.
3. Kord Davis, 'Ethics of Big Data'.
4. Michael Minelli, Michele Chambers, Ambiga Dhiraj, 'Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends'.

LEARNING OUTCOMES:

Upon successful completion of this course, the student will be able to:

1. Model and implement efficient big data solutions for various application areas using appropriately selected algorithms and data structures.
2. Analyse methods and algorithms, to compare and evaluate them with respect to time and space requirements, and make appropriate design choices when solving real-world problems.
3. Motivate and explain trade-offs in big data processing technique design and analysis in written and oral form.
4. Explain the Big Data Fundamentals, including the evolution of Big Data, the characteristics of Big Data and the challenges introduced.
5. Apply non-relational databases, the techniques for storing and processing large volumes of structured and unstructured data, as well as streaming data.
6. Apply the novel architectures and platforms introduced for Big data, in particular Hadoop and MapReduce.

SOFT COMPUTING

Subject Code: BITE2-870 **L T P C** **Duration: 45 Hrs.**
3 1 0 4

COURSE OBJECTIVES:

The objectives of this course are to:

1. To Conceptualize the working of human brain using ANN.
2. To become familiar with neural networks that can learn from available examples and generalize to form appropriate rules for inference systems.
3. To introduce the ideas of fuzzy sets, fuzzy logic and use of heuristics based on human experience.
4. To provide the mathematical background for carrying out the optimization and familiarizing genetic algorithm for seeking global optimum in self-learning situation

Unit-I: Introduction

What is soft computing? Differences between soft computing and hard computing, Soft Computing constituents, Methods in soft computing, Applications of Soft Computing.

Unit-II: Introduction to Genetic Algorithms

Introduction to Genetic Algorithms (GA), Representation, Operators in GA, Fitness function, population, building block hypothesis and schema theorem; Genetic algorithms operators-methods of selection, crossover and mutation, simple GA(SGA), other types of GA, generation gap, steady state GA, Applications of GA

Unit-III: Neural Networks

Concept, biological neural system, Evolution of neural network, McCulloch- Pitts neuron model, activation functions, feedforward networks, feedback networks, learning rules – Hebbian, Delta, Perceptron learning and Windrow-Hoff, winner-take-all.

Unit-IV: Supervised Learning

Perceptron learning, single layer/multilayer perceptron, linear separability, hidden layers, back propagation algorithm, Radial Basis Function network; Unsupervised learning - Kohonen, SOM, Counter-propagation, ART, Reinforcement learning, adaptive resonance architecture, applications of neural networks to pattern recognition systems such as character recognition, face recognition, application of neural networks in image processing.

Unit-V: Fuzzy Systems

Basic definition and terminology, set-theoretic operations, Fuzzy Sets, Operations on Fuzzy Sets, Fuzzy Relations, Membership Functions, Fuzzy Rules & Fuzzy Reasoning, Fuzzy Inference Systems, Fuzzy Expert Systems, Fuzzy Decision Making; Neuro-fuzzy modeling-Adaptive Neuro-Fuzzy Inference Systems, Coactive Neuro-Fuzzy Modeling, Classification and Regression Trees, Data Clustering Algorithms, Rulebase Structure Identification and Neuro-Fuzzy Control, Applications of neuro-fuzzy modeling.

Unit-VI: Swarm Intelligence

What is swarm intelligence? Various animal behavior which have been used as examples, ant colony optimization, swarm intelligence in bees, flocks of birds, shoals of fish, ant-based routing, particle swarm optimization.

Suggested Readings/Books

1. S.N. Shivanandam, 'Principles of Soft Computing', Wiley. ISBN13: 9788126527410, **2011.**
2. Jyh-Shing Roger Jang, Chuen-Tsai Sun, EijiMizutani, 'Neuro-Fuzzy and Soft Computing', Prentice Hall of India, **2003**.
3. George J. Klir and Bo Yuan, 'Fuzzy Sets and Fuzzy Logic-Theory and Applications', Prentice Hall, **1995**.
4. James A. Freeman and David M. Skapura, 'Neural Networks Algorithms, Applications and Programming Techniques', Pearson Edn., **2003**.
5. 5.Mitchell Melanie, 'An Introduction to Genetic Algorithm', Prentice Hall, **1998**.
6. David E. Goldberg, 'Genetic Algorithms in Search, Optimization & Machine Learning', Addison Wesley, **1997**.

LEARNING OUTCOMES:

Upon successful completion of this course, the student will be able to:

1. Ability to analyse and appreciate the applications which can use fuzzy logic.
2. Ability to design inference systems.
3. Ability to understand the difference between learning and programming and explore practical applications of Neural Networks (NN).
4. Ability to appreciate the importance of optimizations and its use in computer engineering fields and other domains.
5. Students would understand the efficiency of a hybrid system and how Neural Network and fuzzy logic can be hybridized to form a Neuro-fuzzy network and its v.