Third Semester (CSE)

Course Code	Type of Course	Course Title	Hours per Week			Marks Distribution		Total Marks	Credits
			L	T	P	Internal	External		
BTCS-18301	Professional Core Course	Digital Electronics	3	0	0	40	60	100	3
BTCS-18302	Professional Core Course	Data structure & Algorithms	3	0	0	40	60	100	3
BTCS-18303	Engineering Science Course	Object Oriented Programming	3	0	0	40	60	100	3
BTAM-18301	Basic Science Course	Mathematics-III	3	1	0	40	60	100	4
BTHS-18901	Humanities & Social Sciences Including Management Course	Fundamentals of Management for Engineers	3	0	0	40	60	100	3
BTCS-18304	Professional Core Course	Digital Electronics Lab	0	0	2	30	20	50	1
BTCS-18305	Professional Core Course	Data structure & Algorithms Lab	0	0	2	30	20	50	1
BTCS-18306	Engineering Science Course	Object Oriented Programming lab.	0	0	2	30	20	50	1
BTCS-18307	Professional Core Course	IT Workshop**	1	0	2	30	20	50	2
BTCS-18308	Internship	Institutional Summer Training*				100	0	100	3
Total			**16**	**0**	**8**	**420**	**380**	**800**	**24**

*This training will be performed by the students in the college in CSE lab for learning Python etc / Central workshop.

** only Practical Examination will be held. No theory Examination is to be held

Fourth Semester(CSE)

Course Code	Type of Course	Course Title	Hours per Week			Marks Distribution		Total Marks	Credits
			L	T	P	Internal	External		
BTAM-18401	Professional Core Course	Discrete Structures	3	1	0	40	60	100	4
BTCS-18402	Engineering Science Course	Computer Organization & Architecture	3	0	0	40	60	100	3
BTCS-18403	Professional Core Course	Operating Systems	3	0	0	40	60	100	3
BTCS-18404	Professional Core Course	Design & Analysis of Algorithms	3	0	0	40	60	100	3
BTHS-18904	Humanities & Social Sciences including Management Course	Organizational Behaviour	3	0	0	40	60	100	3
CSMC-I	Mandatory Course	Environmental Sciences*	-	-	-	-	-	-	0
BTCS-18405	Engineering Science Course	Computer Organization & Architecture Lab	0	0	2	30	20	50	1
BTCS-18406	Professional Core Course	Operating Systems Lab	0	0	2	30	20	50	1
BTCS-18407	Professional Core Course	Design & Analysis of Algorithms Lab	0	0	2	30	20	50	1
Total			**15**	**1**	**6**	**290**	**360**	**650**	**19**

*Students will be given topics related to this course for presentation in the department.

Fifth Semester(CSE)

Course Code	Type of Course	Course Title	Hours per Week			Marks Distribution		Total Marks	Credits
			L	T	P	Internal	External		
BTCS-18501	Engineering Science Course Course	Programming in JAVA	3	0	0	40	60	100	3
BTCS-18502	Professional Core Course	Database Management Systems	3	0	0	40	60	100	3
BTCS-18503	Professional Core Course	Formal Language & Automata Theory	3	0	0	40	60	100	3
BTCS-18504	Professional Core Course	Computer Networks	3	0	0	40	60	100	3
BTHS-18905	Humanities & Social Sciences including Management Course	Effective Technical Communication	3	0	0	40	60	100	3
BTCS-18xxx	Professional Elective	Elective-I	3	0	0	40	60	100	3
CSMC-II	Mandatory Course	Constitution of India/ Essence of Indian Traditional Knowledge*	-	-	-	-	-	-	0
BTCS-18505	Professional Core Course	Database Management Systems Lab	0	0	2	30	20	50	1
BTCS-18506	Professional Core Course	Computer Networks Lab	0	0	2	30	20	50	1
BTCS-18507	Engineering Science	Programming in Java Lab	0	0	2	30	20	50	1
BTCS-18508	Internship	Summer internship					100	100	3
Total			**18**	**0**	**6**	**330**	**520**	**850**	**24**

**Students will be given topics related to this course for presentation in the department.

Sixth Semester(CSE)

Course Code	Type of Course	Course Title	Hours per Week			Marks Distribution		Total Marks	Credits
			L	T	P	Internal	External		
BTCS-18601	Professional Core Course	Compiler Design	3	0	0	40	60	100	3
BTCS-18602	Professional Core Course	Software Engineering	3	0	0	40	60	100	3
BTCS-18xxx	Professional Elective Course	Elective-II	3	0	0	40	60	100	3
BTCS-18xxx	Professional Elective Course	Elective-III	3	0	0	40	60	100	3
BTxx-18xxx	Open Elective Courses	Open Elective-I (Humanities)	3	0	0	40	60	100	3
BTCS-18603	Project	Minor Project	0	0	4	60	40	100	2
BTCS-18604	Professional Core Course	Compiler Design Lab	0	0	2	30	20	50	1
BTCS-18605	Professional Core Course	Software Engineering Lab	0	0	2	30	20	50	1
Total			**15**	**0**	**14**	**320**	**380**	**700**	**19**

Seventh Semester(CSE)

Course Code	Type of Course	Course Title	Hours per Week			Marks Distribution		Total Marks	Credits
			L	T	P	Internal	External		
BTCS-18xxx	Professional Elective Course	Elective-IV	3	0	0	40	60	100	3
BTCS-18xxx	Professional Elective Course Courses	Elective-V	3	0	0	40	60	100	3
BTxx-18xxx	Open Elective Course	Open Elective-II	3	0	0	40	60	100	3
BTCS-18701	Professional Core Course	Artificial Intelligence	3	0	0	40	60	100	3
BTCS-18702	Project	Major Project-I	0	0	8	100	50	150	4
BTCS-18703	Professional Core Course	Artificial Intelligence Lab	0	0	2	30	20	50	1
BTCS-18704	Internship	Summer industrial Training						100	6
Total			**14**	**0**	**10**	**340**	**310**	**750**	**23**

Eighth Semester(CSE)

Course Code	Type of Course	Course Title	Hours per Week			Marks Distribution		Total Marks	Credits
			L	T	P	Internal	External		
BTCS-18xxx	Professional Elective Course	Elective-VI	3	0	0	40	60	100	3
BTxx-18xxx	Open Elective Course	Open Elective-III	3	0	0	40	60	100	3
BTCS-18801	Professional Core Course	Cloud Computing	3	0	0	40	60	100	3
BTCS-18802	Project	Major Project-II	0	0	8	100	50	150	4
BTCS-18803	Internship	*Seminar	2	-	-	50	0	50	2
Total			**11**	**0**	**12**	**270**	**230**	**500**	**15**

***The students will prepare and present seminar based on their Project

LIST OF ELECTIVES

Elective-I

BTCS-18951 Signal & Systems
BTCS-18952 Web Technologies
BTCS-18953 Computational Biology
BTCS-18954 Computer Graphics
BTCS-18955 Python Programming

Elective-II

BTCS-18961 Mobile Application Development
BTCS-18962 Distributed Systems
BTCS-18963 Machine Learning
BTCS-18964 Digital Signal Processing
BTCS-18965 Open Source Technologies

Elective-III

BTCS-18966 Parallel and Distributed Algorithms
BTCS-18967 Embedded Systems
BTCS-18968 Microprocessor and Assembly Language Programming
BTCS-18969 Ad-Hoc and Sensor Networks

Elective-IV

BTCS-18970 Network Security
BTCS-18971 Information Theory and Coding
BTCS-18972 Distributed Operating System
BTCS-18973 Soft Computing
BTCS-18974 Human Computer Interaction

Elective-V

BTCS-18975 Computational Number Theory
BTCS-18976 Speech and Natural Language Processing
BTCS-18977 Parallel Architectures
BTCS-18978 Data Mining
BTCS-18979 Internet Of Things

Elective-VI

BTCS-18981 Queuing Theory and Modeling
BTCS-18982 Real Time Systems
BTCS-18983 Data Analytics
BTCS-18984 Image Processing
BTCS-18985 Information Security & Cyber Law

LIST OF OPEN ELECTIVES

Open electives offered by the department:

Courses of odd semesters:

BTCS-18991 Data Structures & Algorithms

BTCS-18992 Object Oriented Programming

BTCS-18993 Computer Networks

Courses of even semesters:

BTCS-18994 Computer organization & Architecture

BTCS-18995 Operating system

BTCS-18996 Database Management System

THIRD SEMESTER

BTCS-18301
Digital Electronics **L T P**
3 0 0

Internal Marks: 40
External Marks: 60
Total Marks: 100

Objective/s and Expected outcome:
Demonstrate the operation of simple digital gates, identify the symbols, develop the truth table for those gates; combine simple gates into more complex circuits; change binary, hexadecimal, octal numbers to their decimal equivalent and vice versa, demonstrate the operation of a flip-flop. Design counters and clear the concept of shift resisters. Study different types of memories and their applications. Convert digital into analog and vice versa.

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

Boolean Algebra:
Boolean postulates and laws–De-Morgan‟sTheorem, Principle of Duality, Boolean expression – Boolean function, Minimization of Boolean expressions – Sum of Products (SOP), Product of Sums (POS), Min term, Max term, Canonical forms, Conversion between canonical forms, Karnaugh map Minimization, Quine-McCluskey method - Don‟t care conditions.

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.

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, Shift registers.

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

Signal Conversions: Analog & Digital signals. Introduction to A/D and D/A conversion techniques

Suggested Readings/ Books:

1 Morris Mano, Digital Design, Prentice Hall of India Pvt. Ltd
2 Donald P.Leach and Albert Paul Malvino, Digital Principles and Applications, Tata McGraw Hill Publishing Company Limited, New Delhi,

COURSE OUTCOMES (CO): The student is expected to

1. Solve basic binary math operations using the logic gates.
2. Demonstrate programming proficiency using the various logical elements to design practically motivated logical units.
3. Design different units that are elements of typical computer's CPU.
4. Apply knowledge of the logic design course to solve problems of designing of control units of different input/output devices.

BTCS-18302

Data Structure and algorithms

L T P
3 0 0

Internal Marks: 40
External Marks: 60
Total Marks: 100

Objectives of the course:

1 To impart the basic concepts of data structures and algorithms.
2 To understand concepts about searching and sorting techniques
3 To understand basic concepts about stacks, queues, lists trees and graphs.
4 To enable them to write algorithms for solving problems with the help of fundamental Data structures

Detailed contents:

Module1:
Introduction: Basic Terminologies: Elementary Data Organizations, Data Structure Operations: insertion, deletion, traversal etc.; Analysis of an Algorithm, Asymptotic Notations, Time-Space trade off. **Searching:** Linear Search and Binary Search Techniques and their complexity analysis.

Module2:
Stacks and Queues: ADT Stack and its operations: Algorithms and their complexity analysis, Applications of Stacks: Expression Conversion and evaluation – corresponding algorithms and complexity analysis. ADT queue, Types of Queue: Simple Queue, Circular Queue, Priority Queue; Operations on each types of Queues: Algorithms and their analysis.

Module3:
Linked Lists: Singly linked lists: Representation in memory, Algorithms of several operations: Traversing, Searching, Insertion into, Deletion from linked list; Linked representation of Stack and Queue, Header nodes, Doubly linked list: operations on it and algorithmic analysis; Circular Linked Lists: all operations their algorithms and the complexity analysis.

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

Module4:
Sorting and Hashing: Objective and properties of different sorting algorithms: Selection Sort, Bubble Sort, Insertion Sort, Quick Sort, Merge Sort, Heap Sort; Performance and Comparison among all the methods, Hashing.
Graph: Basic Terminologies and Representations, Graph search and traversal algorithms . Minimum Spanning Trees : Prim‟s algorithm, Kruskul‟s algorithm

Suggested books:

1."Fundamentals of Data Structures", Illustrated Edition by Ellis Horowitz, Sartaj Sahni, Computer Science Press.

Suggested reference books:

1 Algorithms, Data Structures, and Problem Solving with C++", Illustrated Edition by Mark Allen Weiss, Addison-Wesley Publishing Company

2 "How to Solve it by Computer", 2nd Impression by R. G. Dromey, Pearson Education.

Course outcomes

1. For a given algorithm student will able to analyze the algorithms to determine the time and computation complexity and justify the correctness.

1 For a given Search problem (Linear Search and Binary Search) student will able to implement it.

2 For a given problem of Stacks, Queues and linked list student will able to implement it and analyze the same to determine the time and computation complexity.

3 Student will able to write an algorithm Selection Sort, Bubble Sort, Insertion Sort, Quick Sort, Merge Sort, Heap Sort and compare their performance in term of Space and Time complexity.

4 Student will able to implement Graph search and traversal algorithms and determine the time and computation complexity.

BTCS-18303
Object Oriented Programming

Objectives of the course
The course will introduce standard tools and techniques for software development, using object oriented approach. To understand Object Oriented Programming concepts and basic characteristics of C++.

Introduction
What is object oriented programming? Procedural Vs. Object-Oriented Programming , Basic Concepts and Principles of OOP

C++ Programming basics
Overview of C++, Program Structure, Exploring the Basic Components of C++ , Type Casting in C++, Operators in C++, Control Structures

Functions
Explore Functions , Describing Call by Value and Call by Reference , Inline Function, Overloading of Functions, String Library Functions, Recursive Functions, Friend Function.

Objects and Classes
Basics of Object and Class, Private and Public Members, Member Functions, Scope Resolution Operator, Constructors and their types, Destructors, Passing Objects as Function Parameters, Returning Objects from Functions.

Inheritance
Concept of inheritance, Derived class and base class, Types of Inheritance, Ambiguity and solution while implementing Multiple Inheritance.

Polymorphism
Concept of Polymorphism, Types of polymorphism, Function Overloading, Operator Overloading, Function Overriding.

Memory Management
Introduction to Pointers, Pointers and Objects, Dynamic Memory Management using new and delete operators, The this Pointer, pointer to object.

Templates and Exception Handling
Concept of Generic Programming, Function Template, Class Template, Exception handling mechanism, use of try, catch and throw keywords

Streams and Files
File Stream Operations, Opening and Closing a File, File Modes, File Pointers, Input Output Operations, Reading/Writing an object into file.

The concepts should be practiced using C++.

Suggested books
1. Lafore R., Object Oriented Programming in C++, Waite Group
2. E. Balagurusamy, Object Oriented Programming with C++, Tata McGraw Hill

Course Outcomes

After taking the course, students will be able to:

1. Recognise features of object-oriented design such as encapsulation, polymorphism, inheritance, and composition of systems based on object identity.
2. Apply these features to program design and implementation.
3. Design applications by using these object oriented concepts.

BTAM-18301

Mathematics – III

3L: 1T: 0P

Objectives

The objective of this course is to familiarize the students with functions of several variables and to introduce effective mathematical tools for the solutions of ordinary differential equations.

Multivariable Calculus (Differentiation) 12 hours

Partial differentiation, total derivative, Composite functions, Implicit functions, Euler‟s Theorem, Jacobians. Applications: Tangent plane and normal line, Taylor‟s and Maclaurin Theorem, Application in estimation of error and approximation, Maxima and Minima, Method of Lagrange‟s multipliers.

Multivariable Calculus (Integration) 10 hours

Double Integral, Change of variable in double integral, Change of order of Integration, Triple Integral, Change of variable in Triple Integral, Applications of Double and Triple Integrals

Ordinary Differential Equations 14 hours

Brief review of first order ordinary differential equations, Exact equations, Equations reducible to exact equations, Equations of the first order and higher degree, Clairaut‟s equation. Linear differential equations with constant co-efficient, Complimentary functions and particular integral, Method of variation of parameters, Equations reducible to linear equations with constant co-efficient (Cauchy‟s and Legendre‟s linear equations), Simultaneous linear equations with constant co-efficient.

Course outcomes

The students will be able to:

- apply mathematical tools of differentiation and integration of functions of multiple variables which are used in various techniques dealing engineering problems.
- understand the methods which can be used to solve the first and higher order ordinary differential equations.

Textbooks/References:

1. Erwin Kreyszig, Advanced Engineering Mathematics, John Wiley & Sons.
2. G.B. Thomas and R.L. Finney, Calculus and Analytic geometry, Pearson.
3. Ramana B.V., Higher Engineering Mathematics, Tata McGraw Hill New Delhi.
4. B.S. Grewal, Higher Engineering Mathematics, Khanna Publishers.
5. S. L. Ross, Differential Equations, Wiley India
6. T. Veerarajan, Engineering Mathematics for first year, Tata McGraw-Hill, New Delhi.

BTHS-18901

Fundamentals of Management for Engineers

L T P
3 0 0

Internal Marks: 40
External Marks: 60
Total Marks: 100

Unit 1. Definition, functions, and significance of Management. Levels of management, Douglas Mc Gregor Theory X and Theory Y. Difference between management and Administration.

Unit 2. Evolution of Management, thought, approaches of management. principles of Henry Fayol and F.W Taylor.

Unit3. Planning and organization nature, objectives and significance of planning, types and steps of planning. Span of control. Methods and types of training, Various organizational structures. Formal and informal organizations.

Unit 4. Concept of motivation, theories of motivation - Maslow need hierarchy theory & Herzberg two factor theory, Concepts of leadership and styles. Steps of Controlling .

Books Recommended:-

1. General Management - C.B. Gupta Sultan Chand

2. Principal and Practice of management- L.M. Prasad Sultan Chand

3. Essential of Management -Koontz & O, Donnel Tata Mc Graw

4. Essential Of Management – Koontz and Weihrich Tata Mc Graw 5. Management : James Stoner, R Edward Freeman, Daniel R. Gilbert, Jr. Prentice Hall of India

BTCS-18304
Digital Electronics Lab

Internal Marks: 30
External Marks: 20

OBJECTIVES : At the end, students should be able to implement 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, Half Adder / Full Adder, Half Subtractor / Full Subtractor, 4-Bit Binary-to-Gray & Gray-to-Binary Code Converter, 4-Bit and 8-Bit Comparator, Multiplexer, Demultiplexer, and Flip Flops.

Implementation all experiments with help of Bread- Board.

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 Subtractor / Full Subtractor: Realization using basic XOR gates.
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.

COURSE OUTCOMES (CO): The student is expected to:

1 Operate laboratory equipment.
2 Construct, analyze, and troubleshoot simple combinational and sequential circuits.
3 Design and troubleshoot a simple state machine.
4. Measure and record the experimental data, analyze the results, and prepare a formal laboratory report.

BTCS-18305

Data Structures Lab

Internal Marks: 30 **L T P**
External Marks: 20 **0 0 2**
Total Marks: 50

List of Practical Exercises, to be implemented using object-oriented approach in C++ Language

1. Write a menu driven program that implements following operations (using separate functions) on a linear array
   - Insert a new element at end as well as at a given position
   - Delete an element from a given array whose value is given or whose position is given
   - To find the location of a given element
   - To display the elements of the linear array
2. Write a menu driven program that maintains a linear linked list and implements the following operations (using separate functions):
   - Insert a new element
   - Delete an existing element
   - Search an element
   - Display all the elements
3. Write a program to demonstrate the use of stack (implemented using linear array) in converting arithmetic expression from infix notation to postfix notation.
4. Program to demonstrate the use of stack (implemented using linear linked lists) in evaluating arithmetic expression in postfix notation.
5. Program to demonstration the implementation of various operations on a linear queue represented using a linear array.
6. Program to demonstration the implementation of various operations on a circular queue represented using a linear array.
7. Program to demonstration the implementation of various operations on a queue represented using a linear linked list (linked queue).
8. Program to illustrate the implementation of different operations on a binary search tree.
9. Program to illustrate the implementation of graph using adjacency matrix.
10. Program to sort an array of integers in ascending order using bubble sort.
11. Program to sort an array of integers in ascending order using selection sort.
12. Program to sort an array of integers in ascending order using insertion sort.
13. Program to sort an array of integers in ascending order using quick sort.
14. Program to demonstrate the use of binary search to search a given element in a sorted array in ascending order.

COURSE OUTCOMES (CO):

The student is expected to:

1. To design and analyze the time and space efficiency of the data structure
2. Identity the appropriate data structure for given problem
3. To have practical knowledge on the application of data structures

BTCS-18306
Object Oriented Programming Lab

Internal Marks: 20
External Marks: 30
Total Marks: 50

1. Introduction to OOP lab (Simple C++ program)
2. WAP to demonstrate the use of Classes and Objects
3. Constructors and Destructors; Write a program to demonstrate different types of constructors and destructors.
4. Operator overloading; Write a program for overloading various unary operators
5. Write a program for overloading various binary operators
6. Memory Management; Write a program to demonstrate the use of new and delete keywords
7. Inheritance; Write a program to demonstrate different types of inheritance
8. Write a program to remove ambiguity from hybrid inheritance
9. Polymorphism; Write a program for polymorphism(virtual function)
10. Write a program for templates (class and function template)
11. File handling; Write a program to copy contents of one file to another file.
12. Program using streams

Course outcomes:

The student is expected to:
1. Conceptualize the given problem and transform it in to an Object Oriented system.
2. Implement coding standard and verification practices
3. Build expertise in Object Oriented programming language.

BTCS-18307
IT Workshop

L T P
1 0 2

Objective of Course: The objective of this course is to demonstrate the students, the basic features of MATLAB/SCILAB and hands on training on any of them so that the students could be able to use this tool later on for various projects and thesis work.

Basics of MATLAB/SCILAB : Introduction, Basic features, A minimum session, Starting MATLAB/Scilab, using MATLAB/Scilab as a calculator , Quitting MATLAB/SCILAB , creating variables, Overwriting variable, Error messages, Making corrections, Controlling the hierarchy of operations or precedence , Controlling the appearance of floating point number , Managing the workspace, Entering multiple statements per line.

Basic Graphics and Matrix Algebra: Mathematical functions, Basic plotting, overview, Creating simple plots , Adding titles, axis labels, and annotations , multiple data sets in one plot, Specifying line styles and colors, Entering a vector, Entering a matrix, Matrix indexing, Colon operator, Linear spacing, Colon operator in a matrix, creating a sub-matrix, Deleting row or column, Transposing a matrix , Concatenating matrices.

Array operations and Linear equations: Array operations, Matrix arithmetic operations , Array arithmetic operations , Solving linear equations , Matrix inverse

Introduction to programming in MATLAB/SCILAB : Introduction , M-File Scripts, Examples, Script side-effects , File functions , Anatomy of a M-File function, Input and output arguments , Input to a script file , Output commands

Control flow and operators: Introduction , Control flow, The „„if...end““ structure, Relational and logical operators , The„„for...end““loop , The „„while...end““loop Other flow structures, Saving output to a file.

List of Experiments: (using MATLAB or SCILAB)

1. Installation of MATLAB/SCILAB.
2. Various Operations of mathematics in MATLAB/SCILAB.
3. Handling of matrices.
4. To plot different types of two dimensional plots.
5. Branching statments-IF,IF-ELSE,Switch.
6. Write a program for printing below mention pattern.

```
    1
  1   2
1   2   3
1   2   3   4
```

7. Write a program for calculating whether number is prime or not with the help of function.
8. Write a program for implementation of basic Calculator.

Text Book:

1. Stephen J. Chapman . MATLAB- Programming for Engineers, Fourth Edition 2008
2. Holly More.MATLAB-for Engineers,Fourth Edition by Pearson.

Expected course outcomes: After the course, students will be

1. Able to use Matlab/Scilab for interactive computations.
2. Able to generate plots and export this for use in reports and presentations.
3. Able to program scripts and functions using the Matlab/Scilab development environment.
4. Able to use basic flow controls (if-else, for, while).
5. Familiar with strings and matrices and their use.

FOURTH SEMESTER

BTAM-18401

Discrete Structures (CSE & IT) 3L: 1T: 0P

Objectives

To provide knowledge of combinatorial problems, algebraic structures and graph theory required for building mathematical foundation of computer science.

Sets, Relations and Functions 6 hours

Basic operations and laws on sets, Cartesian products, Binary relation, Partial order relation, Equivalence Relation, different types of functions, their compositions and inverses.

Propositional Logic 6 hours

Syntax and semantics, proof systems, satisfiability, validity, soundness, completeness, deduction theorem, etc. Decision problems of propositional logic. Introduction to first order logic and first order theory.

Partially ordered sets 6 hours

Complete partial ordering, chain, lattice, complete, distributive, modular and complemented lattices, Boolean and pseudo Boolean lattices.

Introduction to Counting 6 hours

Basic counting techniques – inclusion and exclusion, pigeonhole principle, permutation, combination, Introduction to recurrence relation and generating functions.

Algebraic Structures 8 hours

Algebraic structures with one binary operation – semigroup, monoid and group. Cosets, Lagrange‟s theorem, normal subgroup, homomorphic subgroup. Congruence relation and quotient structures. Algebraic structures with two binary operations- ring, integral domain, and field. Boolean algebra and Boolean ring (Definitions and simple examples only).

Introduction to Graphs 6 hours

Graphs and their basic properties – degree, path, cycle, subgraph, isomorphism, Eulerian and Hamiltonian walk, trees.

Course Outcomes

Students will be able to:

- express logic sentence in terms of predicates, quantifiers, and logical connectives.
- derive the solution using deductive logic and prove the solution
- classify the algebraic structures of mathematical problems
- to evaluate Boolean functions and simplify expressions using the properties of Boolean Algebra
- to develop the given problem as graph networks and solve with techniques of graph theory.

Textbooks/References:

J. P. Tremblay and R. P. Manohar, Discrete Mathematics with Applications to Computer Science, Tata McGraw-Hill
C.L. Liu, Elements of Discrete Mathematics, Tata McGraw-Hill
R. C. Penner, Discrete Mathematics: Proof Techniques and Mathematical Structures, World Scientific
R. L. Graham, D. E. Knuth, and O. Patashnik, Concrete Mathematics, Addison-Wesley
K. H. Rosen, Discrete Mathematics and its Applications, Tata McGraw-Hill
J. L. Hein, Discrete Structures, Logic, and Computability, Jones and Bartlett
Narsingh Deo, Graph Theory with Application to Engineering and Computer Science, PHI

BTCS-18402
Computer Organization & Architecture

L:3 T:0 P:0

Objectives of the course:

To expose the students to the following:

1 Understanding of its various functional units of computer system.
2 Microprocessors, instruction execution and assembly level programming.
3 Fast Adders, Multiplication and Division Algorithms.
4 Control Unit Microprogramming, Hardwired control unit
5 Memory hierarchy, mapping and memory system design
6 IO Modes: Program control IO, DMA, Interupt initiated IO.
7 Basic concepts of pipelining, parallel processors and cache coherency.

Functional blocks of a computer: CPU, memory, input-output subsystems, control unit. Microprocessor based system design, Introduction to 8085 architecture, addressing modes, instruction set and instruction execution cycle.

Data Representation and Binary Arithmetic: Signed number representation, addition, subtraction, Booth multiplication algorithm, division algorithms, ripple adder, carry look ahead adders and array multipliers.

CPU control unit design: Hardwired and micro-programmed design approaches, Control Memory, RISC/CISC architecture.

Memory system design: Memory Hierarchy, memory organization, interleave memory, virtual memory, cache memory mapping techniques, and replacement algorithms, write policies.

IO Modes and Interfaces: Input-output subsystems, I/O device interface, I/O transfers – program controlled, interrupt driven and DMA, privileged and non-privileged instructions, software interrupts and exceptions. I/O device interface – ATA, SCSI, USB. Hard Disk Drive construction and working.

Pipelining: Basic concepts of pipelining, speedup efficiency and throughput, pipeline hazards.

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

Suggested books:

1. "Computer System Architecture", 3rd Edition by M. Morris Mano, Pearson Education India.
2. "Fundamentals of Microprocessor and Microcontrollers", by B Ram, Dhanpat Rai Publications.
3. "Computer Architecture and Organization", 3rd Edition by John P Hayes, McGraw Hill Education.
4. "Computer Organization and Architecture: Designing for Performance", 10th Edition by William Stallings, Pearson Education.

Course outcomes:

1. Draw the functional block diagram of bus architecture of a computer and describe the function of the instruction execution cycle, interpretation of instructions, addressing modes.
2. Implement assembly language program for given task like computing addition, subtraction, multiplication, division, searching, sorting etc.
3. Categorize memory organization and explain the function of each element of a memory hierarchy.
4. Identify and compare different methods for computer I/O mechanisms.
5. Write a flowchart for Concurrent access to memory and cache coherency in Parallel Processors and describe the process.

BTCS-18403

Operating Systems **L T P**
3 0 0

Objectives of the course

To learn the fundamentals of Operating Systems.

1. To learn the mechanisms of OS to handle processes and threads and their communication;
2. To learn the mechanisms involved in memory management in contemporary OS;
3. To gain knowledge on distributed operating system concepts that includes architecture, Mutual exclusion algorithms, deadlock detection algorithms and agreement protocols ; and
4. To know the components and management aspects of concurrency management.

Module 1:

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

Module 2:

Processes: Definition, Process Relationship, Different states of a Process, Process State transitions, Process Control Block (PCB), Context switching
Thread: Definition, Various states, Benefits of threads, Types of threads, Concept of multithreads,
Process Scheduling: Foundation and Scheduling objectives, Types of Schedulers, Scheduling criteria: CPU utilization, Throughput, Turnaround Time, Waiting Time, Response Time; Scheduling algorithms: Pre-emptive and Non pre-emptive, FCFS, SJF, RR; Multiprocessor scheduling: Real Time scheduling: RM and EDF.
Module 3:
Inter-process Communication: Critical Section, Race Conditions, Mutual Exclusion, Hardware Solution, Strict Alternation, Peterson‟s Solution, The Producer\ Consumer Problem, Semaphores, Event Counters, Monitors, Message Passing, Classical IPC Problems: Reader‟s & Writer Problem, Dinning Philosopher Problem etc.

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

Module 5:
Memory Management: Basic concept, Logical and Physical address map, Memory allocation: Contiguous Memory allocation – Fixed and variable partition–Internal and External fragmentation and Compaction; Paging: Principle of operation – Page allocation – Hardware support for paging, Protection and sharing, Disadvantages of paging.
Virtual Memory: Basics of Virtual Memory – Hardware and control structures – Locality of reference, Page fault , Working Set , Dirty page/Dirty bit – Demand paging, Page Replacement algorithms: Optimal, First in First Out (FIFO), Second Chance (SC), Not recently used (NRU) and Least Recently used (LRU).

Module 6:
I/O Hardware: I/O devices, Device controllers, Direct memory access Principles of I/O Software: Goals of Interrupt handlers, Device drivers, Device independent I/O software, Secondary-Storage Structure: Disk structure, Disk scheduling algorithms
File Management: Concept of File, Access methods, File types, File operation, Directory structure, File System structure, Allocation methods (contiguous, linked, indexed), Free-space management (bit vector, linked list, grouping), directory implementation (linear list, hash table), efficiency and performance.

Module 7:
Disk Management: Disk structure, Disk scheduling - FCFS, SSTF, SCAN, C-SCAN, Disk reliability, Disk formatting, Boot-block, Bad blocks

Suggested books:

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

Suggested reference books:

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

Course Outcomes:

1. Create processes and threads;
2. Develop algorithms for process scheduling for a given specification of CPU utilization, Throughput, Turnaround Time, Waiting Time, Response Time.
3. For a given specification of memory organization develop the techniques for optimally allocating memory to processes by increasing memory utilization and for improving the access time.
4. Design and implement file management system.
5. For a given I/O devices and OS (specify) develop the I/O management functions in OS as part of a uniform device abstraction by performing operations for synchronization between CPU and I/O controllers.

BTCS-18404

Design and Analysis of algorithm

L T P
3 0 0

Objectives of the course

1. Analyze the asymptotic performance of algorithms.
2. Write rigorous correctness proofs for algorithms.
3. Demonstrate a familiarity with major algorithms and data structures.
4. Apply important algorithmic design paradigms and methods of analysis.
5. Synthesize efficient algorithms in common engineering design situations.

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

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

Module 3:
Graph and Tree Algorithms: Traversal algorithms: Depth First Search (DFS) and Breadth First Search (BFS); Shortest path algorithms, Transitive closure, Minimum Spanning Tree, Topological sorting, Network Flow Algorithm.

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

Module 5:
Advanced Topics: Approximation algorithms, Randomized algorithms, Class of problems beyond NP – P SPACE

Suggested books:
1 Introduction to Algorithms, 4TH Edition, Thomas H Cormen, Charles E Lieserson, Ronald L Rivest and Clifford Stein, MIT Press/McGraw-Hill.
2 Fundamentals of Algorithms – E. Horowitz et al.

Suggested reference books
1 Algorithm Design, 1ST Edition, Jon Kleinberg and ÉvaTardos, Pearson.
2 Algorithm Design: Foundations, Analysis, and Internet Examples, Second Edition, Michael T Goodrich and Roberto Tamassia, Wiley.

Course Outcomes

1. For a given algorithms analyze worst-case running times of algorithms based on asymptotic analysis and justify the correctness of algorithms .
2. Describe the greedy paradigm and explain when an algorithmic design situation calls for it. For a given problem develop the greedy algorithms.
3. Describe the divide-and-conquer paradigm and explain when an algorithmic design situation calls for it. Synthesize divide-and-conquer algorithms. Derive and solve recurrence relation.
4. Describe the dynamic-programming paradigm and explain when an algorithmic design situation calls for it. For a given problems of dynamic-programming and develop the dynamic programming algorithms, and analyze it to determine its computational complexity.
5. For a given model engineering problem model it using graph and write the corresponding algorithm to solve the problems.
6. Explain the ways to analyze randomized algorithms (expected running time, probability of error).

BTHS-18904

Organisational Behaviour

Credit:3

L T P

3 0 0

Internal Marks: 40

External Marks: 60

Total Marks: 100

UNIT 1:

Introduction to Organisational Behaviour: the OB Model, Roles of Manager in OB, Douglas McGregor Theory X and Theory Y.

UNIT2:

Foundation of Individual Behaviour: Concepts of Motivation, Personality, Values, Attitudes, Perception, Learning, Individual Decision-Making and Problem-Solving.

UNIT3:

Foundation of Group Behaviour: Concepts related to Communication, Concept of leadership and styles, Work Teams and Group Dynamics.

UNIT4:

Foundation of the Organisation: Concepts related to Organisation Structure, Organization Culture, Organizational Conflict and Discipline.

UNIT 5:

Organisation Management: Definition of management, Function of Management, Maslow Hierarchy, Principles of Henry Fayol and F.W Taylor.

Suggested Reading:

Organizational Behaviour – Stephen P. Robbins, Timothy A.Judge, SeemaSanghi

Organisational Behaviour – L M Prasad

Organizational Behavior, Human Behavior At Work – John W Newstorm

Management &Organisational Behaviour – Laurie J Mulli

BTCS-18405
Computer Organization & Architecture Lab

L:2 T:0 P:2

Objective: To understand the basic concepts of Computer Organization and Assembly Language Programming.

1. Familiarization with Computer Hardware and Peripherals.
2. Familiarization with 8085 Microprocessor kit.
3. Write an ALP to add of two 8 bit numbers to get 8 bit sum.
4. Write a ALP to add two 8 bit numbers to get 16 bit sum.
5. Write an ALP to add two 16-bit numbers to get 16-bit sum.
6. Write an ALP to add two 8-bit BCD Numbers.
7. Write an ALP to exchange the contents of BC pair with DE Pair.
8. Write an ALP to find 1"s complement and 2"s complement of an 8 bit number.
9. Write an ALP to subtract two signed 8-bit numbers.
10. Write an ALP to find sum of "n" 8 bit numbers.
11. Write an ALP to multiply an 8 bit number by 2 using shifting method.
12. Write an ALP to find larger of two 8 bit numbers.
13. Write an ALP to find largest number among "n" 8-bit numbers.
14. Write an ALP to sort "n" numbers in ascending order.

Course Outcome (CO):

1. To get familiar with various parts of computer system and peripherals..
2. To get familiar with Microprocessor, PIN structure, use of opcodes, kit details etc.
3. To know the use of various instructions, stack pointer and flags etc.
4. Write ALPs to do logical, arithmetic, searching and sorting operations.

BTCS-18406
Operating System Lab

Objective: This course provides knowledge of different operating systems.

1. Installation and Configuration of latest Windows Operating System;
2. Installation and Configuration of latest UNIX Operating System (any flavor)
3. Practice of using various editors for example vim, ex, and ed;
4. Practice of commands used in networking for example ifconfig, ipconfig, traceroute, telnet, nslookup, netstat, scp, nmap, ping;
5. Practice of directory listing commands with various options for example ls, dir
6. Practice of process related commands for example, ps, who, kill, sleep;
7. Practice of backup and recovery commands for example tar, cpio;
8. Practice of file management commands for example cd, cp, rm, mkdir, rmdir, cat, ws, sat, cut, grep, chmod,, chown, dd, df;
9. Printing commands, grep, fgrep, find, sort, cal, banner, touch file,
10. Shell Programming: Basic of shell programming, various types of shell, Shell Programming in bash, conditional & looping statement, case statements, parameter passing and arguments, shell variables, shell keywords, creating shell programs for automate system tasks, report printing.
11. Virtualization, Installation of Virtual Machine Software such as vmware / virtualbox and installation of various operating System on Virtual Machine;

COURSE OUTCOMES (CO): The expected outcomes are:

1. Installation and configuration of Windows and UNIX Operating Systems.
2. Practice of various commands in Windows and Unix Operating Systems
3. Learning of Shell Programming; and
4. Knowledge of virtualization.

BTCS-18407
Design & Analysis of Algorithms Lab

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

1. Code and analyze to compute the greatest common divisor (GCD) of two numbers.

2. Code and analyze to find the median element in an array of integers.

3. Code and analyze to find the majority element in an array of integers.

4. Code and analyze to sort an array of integers using Heap sort.

5. Code and analyze to sort an array of integers using Merge sort.

6. Code and analyze to sort an array of integers using Quick sort.

7. Code and analyze to find the edit distance between two character strings using dynamic programming. 8. Code and analyze to find an optimal solution to weighted interval scheduling using dynamic programming.

9. Code and analyze to find an optimal solution to matrix chain multiplication using dynamic programming.

10. Code and analyze to do a depth-first search (DFS) on an undirected graph. Implementing an application of DFS such as (i) to find the topological sort of a directed acyclic graph, OR (ii) to find a path from source to goal in a maze.

11. Code and analyze 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 analyze to find the minimum spanning tree in a weighted, undirected graph.

13. Code and analyze to find all occurrences of a pattern P in a given string S.

14. Code and analyze to multiply two large integers using Karatsuba algorithm.

15. Code and analyze to compute the convex hull of a set of points in the plane.

16. (Mini-project Topic) Program to multiply two polynomials using Fast Fourier Transform.

COURSE OUTCOMES (CO): The students are expected to:

1. Write code for different methods of computation;

2. Practice of writing program different searching and sorting algorithms and do analysis of the same; and

3. Coding in dynamic programming.

FIFTH SEMESTER

BTCS-18501 Programming in Java

L T P
3 1 0

Internal Marks: 40
External Marks: 60
Total Marks: 100

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. This subject will help to improve the analytical skills of object oriented programming.

1. Overview of Java: Object oriented programming, paradigms, abstraction, OOP principles, Java class libraries, Date types, Variables and Arrays: Integers, floating-point types, characters, Boolean, Iterates, Variable, Data types and casting, automatic type promotion in expressions, arrays. [9]

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

3. Introduction to Classes: Class fundamentals, declaring object reference variable, Introducing methods, constructors, this keyword, garbage collection, the finalize () method. [2]

4. Methods and Classes: Overloading methods, using objects as parameters, recursion. [3]

5. Inheritance: Inheritance basics and it types, using super, method overriding, Package and Interfaces, Package access protection, importing packages. [4]

6. 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, creating your own exception sub classes, using exceptions. [4]

7. 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. [5]

8. 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. [4]

9. RW Files and Applets: Reading and Writing Files, Applet Fundamentals, Applet Architecture, The HTML Applet tag, Passing parameters to Applets. [3]

10. Networking: TCP/IP Client Sockets URL, URL Connection, TCP/IP Server Sockets, Database connectivity. [3]

Course Outcomes:

1. Understand the concepts of OOP & Java language.

2. Show competence in the use of the Java programming language in the development of small to medium-sized application programs that demonstrate professionally acceptable coding and performance standard
3. Participate and succeed in competitive examinations like GATE, Engineering services, recruitment interviews etc.
4. Plan their career in java based technologies like HADOOP etc.

Suggested Books:

1. D.S. Bedi, Strength of Materials, Khanna Book Publishing Company.
2. E.P. Popov, Mechanics of Materials-(SI Version), Prentice Hall India.
3. R.S Lehri and A.S. Lehri, Strength of Materials, Kataria and Sons.
4. S.S.Rattan, Strength of Materials, Tata McGraw Hill.

BTCS-18502 Database Management Systems

L T P
3 0 0

Internal Marks: 40
External Marks: 60
Total Marks: 100

Course Objectives:

1. To understand the different issues involved in the design and implementation of a database system.
2. To study the physical and logical database designs, database modeling, relational, hierarchical, and network models
3. To understand and use data manipulation language to query, update, and manage a database
4. To develop an understanding of essential DBMS concepts such as: database security, integrity, concurrency, distributed database, and intelligent database, Client/Server (Database Server), Data Warehousing.
5. To design and build a simple database system and demonstrate competence with the fundamental tasks involved with modeling, designing, and implementing a DBMS

Detailed contents:

Module 1

Database system architecture: Data Abstraction, Data Independence, Data Definition Language (DDL), Data Manipulation Language (DML).

Data models: Entity-relationship model, network model, relational and object oriented data models, integrity constraints, data manipulation operations.

Module 2:

Relational query languages: Relational algebra, Tuple and domain relational calculus, SQL3, DDL and DML constructs, Open source DBMS - MYSQL.

Relational database design: Domain and data dependency, Armstrong's axioms, Normal forms, Dependency preservation, Lossless design.

Query processing and optimization: Evaluation of relational algebra expressions, Query equivalence, Join strategies, Query optimization algorithms.

Module 3:

Storage strategies: Indices, B-trees, hashing.

Module 4:

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

Module 5:

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

Module 6:

Advanced topics: Object oriented and object relational databases, Logical databases, Web databases, Distributed databases, Data warehousing and data mining.

Course Outcomes

1. For a given query write relational algebra expressions for that query and optimize the developed expressions

2.For a given specification of the requirement design the databases using E_R method and normalization.

3. For a given specification construct the SQL queries for Open source DBMS –MYSQL

4. For a given query, optimize its execution using Query optimization algorithms

5. For a given transaction-processing system, determine the transaction atomicity, consistency, isolation, and durability.

6. Understand the isolation property, including locking, time stamping based on concurrency control and Serializability of scheduling.

Suggested books:

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

Suggested reference books

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

BTCS-18503 Formal Language & Automata Theory

Internal Marks: 40 **L T P**
External Marks: 60 **3 1 0(Credits 3)** **Total Marks: 100**

Course Objectives: The course is designed to understand the basic concepts to develop a formal notation for strings, languages and machines, Design finite automata to accept a set of strings of a language, Prove that a given language is regular and apply the closure properties of languages, Design context free grammars to generate strings from a context free language and convert them into normal forms, Prove equivalence of languages accepted by Push Down Automata and languages generated by context free grammars, Identify the hierarchy of formal languages, grammars and machines, Distinguish between computability and non-computability and Decidability and undecidability.

Module 1:

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

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

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

Context-sensitive languages: Context-sensitive grammars (CSG).

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

Undecidability: Church-Turing thesis, universal Turing machine, the universal and diagonalization languages. Rice s theorem, undecidable problems about languages.

Course Outcomes:

1. Write a formal notation for strings, languages and machines.
2. Design finite automata to accept a set of strings of a language.
3. For a given language determine whether the given language is regular or not.
4. Design context free grammars to generate strings of context free language.
5. Determine equivalence of languages accepted by Push Down Automata and languages generated by context free grammars.
6. Write the hierarchy of formal languages, grammars and machines.

7. Distinguish between computability and non-computability and Decidability and undecidability.

Suggested Books:

1. John E. Hopcroft, Rajeev Motwani and Jeffrey D. Ullman, Introduction to Automata Theory, Languages, and Computation, Pearson Education Asia.
2. K.L.P. Mishra and N. Chandrasekaran, "Theory of Computer Science, Third Edition", PHI Learning Private Limited, 2011.

Suggested reference Books:

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

BTCS-18504 Computer Networks

Internal Marks: 40	**L T P**
External Marks: 60	**3 0 0**
Total Marks: 100	

Objectives of the course:-

- To develop an understanding of network architectures from the design and performance perspective.
- To introduce the students to the major concepts involved in local area networks (LANs), wide-area networks (WANs) and Wireless LANs (WLANs).
- To understand the requirement and use of various protocols at different levels of computer network architecture.
- To apply the knowledge of different network designs and various logical models of networking to solve problems of communication.
- To understand the basic concepts related to network security, firewall and cryptography.

Detailed contents:-

Data Communication Components:- Representation of data and data flow. Uses of computer networks. OSI and TCP/IP reference models. Various transmission media types. Network Topologies, Protocols and Standards, Wired and Wireless Networks. Techniques for Bandwidth utilization: Multiplexing- Frequency division, Time division.

Data Link Layer:- Fundamentals of Error Detection and Error Correction. Block coding, Hamming Code and CRC. Flow Control and Error control protocols - Stop and Wait, Sliding Window: Go back – N ARQ, Selective Repeat ARQ. Piggybacking.

Medium Access Sub Layer:- MAC Address. Multiple access protocols - Pure ALOHA, Slotted ALOHA, CSMA, CSMA/CD, and CDMA/CA. Binary exponential backoff algorithm. Collision free protocols.

Network Layer:- Logical addressing: IPV4, IPV6, CIDR, Subnet Mask, Default Gateway and DHCP. Circuit and Packet Switching. Routing algorithms: Flooding, Distance Vector, Link State. Congestion control policies, Leaky bucket and token bucket algorithms.

Transport Layer:- Port numbers and socket address. Process to Process Communication. Connection less and Connection oriented services. Transmission Control Protocol (TCP), User Datagram Protocol (UDP).

Application Layer:- Domain Name System (DNS), TELNET, EMAIL, File Transfer, Protocol (FTP), WWW, HTTP, SNMP, Firewalls, Basic concepts of Cryptography.

Suggested books:-

1. Data Communication and Networking, 4th Edition, Behrouz A. Forouzan, McGraw- Hill.
2. Computer Networks, 4th Edition, Andrew S. Tanenbaum, Pearson Education.

Reference Books:-

1. Internetworking with TCP/IP, Volume-I, Douglas E. Comer, Pearson Education.
2. Data and Computer Communication, 8th Edition, William Stallings, Pearson Education.
3. Computer Networking: A Top-Down Approach, James F. Kurose & Keith W. Ross, Pearson Education.

Course Outcomes:-

1. Explain the functions of the different layers of the OSI and TCP/IP Reference Models.
2. Understand the major concepts and devices involved in LAN and WAN.
3. Need and use of various error detection and correction methods.
4. Apply algorithms for medium access sub layer for maximum utilization of the bandwidth.
5. Configure networks, use of IP, MAC addresses, subnet mask, default gateway, ping and tracert.
6. Familiarization with application layer protocols: DNS, TELNET, FTP and HTTP etc.
7. Understand the importance of network security, firewall and cryptography.

BTHS-18905 Effective Technical Communication

L:3, T:,P:0

Module 1: Information Design and Development- Different kinds of technical documents, Information development life cycle, Organization structures, factors affecting information and document design, Strategies for organization, Information design and writing for print and for online media.

(7hrs)

Module 2: Technical Writing, Grammar and Editing- Technical writing process, forms of discourse, Writing drafts and revising, Collaborative writing, creating indexes, technical writing style and language. Basics of grammar, study of advanced grammar, editing strategies to achieve appropriate technical style. Introduction to advanced technical communication, Usability, Hunan factors, Managing technical communication projects, time estimation, Single sourcing, Localization.

(7hrs)

Module 3: Self Development and Assessment- Self assessment, Awareness, Perception and Attitudes, Values and belief, Personal goal setting, career planning, Self-esteem. Managing Time; Personal memory, Rapid reading, Taking notes; Complex problem solving; Creativity.

(7hrs)

Module 4: Communication and Technical Writing- Public speaking, Group discussion, Oral; presentation, Interviews, Graphic presentation, Presentation aids, Personality Development. Writing reports, project proposals, brochures, newsletters, technical articles, manuals, official notes, business letters, memos, progress reports, minutes of meetings, event report.

(7hrs)

Module 5: Ethics- Business ethics, Etiquettes in social and office settings, Email etiquettes, Telephone Etiquettes, Engineering ethics, Managing time, Role and responsibility of engineer, Work culture in jobs, Personal memory, Rapid reading, Taking notes, Complex problem solving, Creativity.

(7hrs)

Text/Reference Books:

1. David F. Beer and David McMurrey, Guide to writing as an Engineer, John Willey. New York, 2004
2. Shiv Khera, You Can Win, Macmillan Books, New York, 2003.
3. Raman Sharma, Technical Communications, Oxford Publication, London, 2004.
4. Sharma, R. and Mohan, K. Business Correspondence and Report Writing, TMH New Delhi 2002.

BTCS-18505 Database Management Systems Lab

Internal Marks: 30 **L:0 T:0 P:2**
External Marks: 20
Total Marks: 50

Course Objectives: This practical will enable students to:

- Familiarize with different DBMS.
- Create and use a database.
- Familiarized with a query language.
- Have hands on experience on DDL Commands.
- Have a good understanding of DML Commands and DCL commands.
- Familiarize advanced SQL queries.
- Expose to different applications.

List of experiments:

1. Installation and configuration of Postgresql/Mysql/Oracle/SQL Server
2. Introduction to Structured Query Languages (SQL)
   a. Use of CREATE, ALTER and DROP DATABASE Commands;
   b. Use of CREATE, ALTER and DROP TABLE Commands;
   c. Use of Insert, update and delete commands;
   d. Use of Select statement.
3. Use of Primary Key, Foreign Key and other Constraints.
4. Use of CURSOR statement.
5. Use of TRIGGER statement.
6. Use of FUNCTIONS and STORED PROCEDURES.
7. Use of CREAE USER, ALTER USER and DROP USER Command.
8. Use of GRANT and REVOKE statements.
8. Creation of Views, Synonyms, Sequence, Indexes, Save point
9. Mini project (Application Development using Postgresql/Mysql/Oracle/SQL Server)
   a) Employee Management System;
   b) Hospital Management System;
   c) Railway Reservation System.

Course Outcomes: At the end of the course, the student should be able to: Design and implement a database schema for a given problem-domain; Populate and query a database.

Suggested Reading:

1. Documentation on Oracle at Oracle website.
2. Documentation on SQL Server at Microsoft website
3. Online tutorials available on PostgreSQL and MySQL

BTCS-18506 Computer Networks Lab

Internal Marks: 30 **L:0 T:0 P:2**
External Marks: 20
Total Marks: 50

Objective: To Understand The Concepts of Computer Networking.

List of Practicals:-

1. Familiarization with the latest computer hardware and operating systems.
2. Introduction to Networking Components and devices: Network Interface Controllers, Hubs, Switches, Routers etc.
3. Study of various transmission cables and Tools: Co-axial cable, UTP Cable, Crimping Tools.
4. Preparing straight through and crossover UTP cables.
5. Configuration of IP Addresses, Subnet Mask and default gateway in Windows and Linux.
6. Implementation of file and printer sharing.
7. Designing and implementing IPv4 Class A, B and C Networks.
8. Troubleshooting with the use of ipconfig, ping and traceroute commands.
9. Familiarization with Remote Desktop Connection, FTP, HTTP and Telnet.

Course Outcome (CO):-

1. Know the various latest hardware components and networking devices.
2. Use of different types of transmission media.
3. Understand different networking topologies.
4. Configure networks and use of various troubleshooting commands.
5. Use of different application layer protocols.

BTCS-18507 Programming in Java Lab

	L	T	P
Internal Marks: 30			
External Marks: 20	**0**	**0**	**2**
Total Marks: 50			

1. Install JDK, Write a simple "Hello World" or similar java program, compilation, debugging, executing using java compiler and interpreter.
2. Write a program in Java to find maximum of three numbers using conditional operator.
3. Write a program to find the largest numbers among 10 numbers in Array.
4. Write a program in Java to multiply two matrix.
5. Write a program in Java to demonstrate single inheritance, multilevel inheritance and hierarchical inheritance.
6. Write a program in Java to demonstrate implementation of multiple inheritance using interfaces.
7. Write a program in Java in which a subclass constructor invokes the constructor of the super class and instantiate the values.
8. Write a program to implements the exception handling.
9. Write a program in Java to demonstrate multiple try block and multiple catch exception.
10. Write a program to implements the concept of Multithreads.
11. Write a program that executes two threads. One thread will print the even numbers and the thread will print odd numbers from 1 to 50.
12. Write a program to show the concept of Packages.
13. Write a program for Applets.
14. Write a program to implement of mouse events and keyboard events.
15. Write a program to implement basic file reading and writing methods.
16. Write a program to connect to Database using JDBC.

Elective – I

BTCS-18951 Signals and System

Internal Marks : 40 **L T P**
External Marks : 60 **3 0 0**
Total Marks : 100

Signals and systems as seen in everyday life, and in various branches of engineering and science.

Energy and power signals, continuous and discrete time signals, continuous and discrete amplitude signals. System properties: linearity: additivity and homogeneity, shift-invariance, causality, stability, realizability, LTI systems. (8 hours)

Linear shift-invariant (LSI) systems, impulse response and step response, convolution, input- output behavior with aperiodic convergent inputs. Characterization of causality and stability of linear shift-nvariant systems. System representation through differential equations and difference equations. (8 hours)

Periodic and semi-periodic inputs to an LSI system, the notion of a frequency response and its relation to the impulse response, Fourier series representation, the Fourier Transform, convolution/multiplication and their effect in the frequency domain, magnitude and phase response, Fourier domain duality. The Discrete-Time Fourier Transform (DTFT) (8 hours)

The z-Transform for discrete time signals and systems- eigen functions, region of convergence, z-domain analysis. (4 hours)

The Sampling Theorem and its implications- Spectra of sampled signals. Reconstruction: ideal interpolator, zero-order hold, first-order hold. Aliasing and its effects. (8 hours)

Text/Reference books:

1. A.V. Oppenheim, A.S. Willsky and I.T. Young, "Signals and Systems", Prentice Hall,
2. R.F. Ziemer, W.H. Tranter and D.R. Fannin, "Signals and Systems - Continuous and Discrete", 4th edition, Prentice Hall, 1998.
3. Papoulis, "Circuits and Systems: A Modern Approach", HRW, 1980.
4. B.P. Lathi, "Signal Processing and Linear Systems", Oxford University Press, c1998.
5. Douglas K. Lindner, "Introduction to Signals and Systems", McGraw Hill International Edition: c1999.

Course outcomes: At the end of this course students will demonstrate the ability to

1. Analyze different types of signals
2. Represent continuous and discrete systems in time and frequency domain using different transforms
3. Investigate whether the system is stable
4. Sampling and reconstruction of a signal

BTCS-18952 Web Technologies

L T P
3 0 0

Internal Marks: 40
External Marks: 60
Total Marks: 100

Course Objectives: This Subject is useful for Making own Web page and how to host own web site on internet Along with that Students will also learn about the protocols involve in internet technology.

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]

2. **HTML5 and CSS:** Introduction to HTML, Lists, adding graphics to HTML page, creating tables, linking documents, forms, DHTML ,Cascading Style sheets and Responsive Web Design. [7]

3. **JAVASCRIPT:** Introduction, programming constructs: variables, operators and expressions, conditional checking, functions and dialog boxes, JavaScript DOM, creating forms, introduction to Cookies,Lightbox. [5]

4. **INTRODUCTION:** Introduction to JQuery, JSON, AngularJS, NodeJS, Wordpress, DSN,API's [8]

5. **BOOTSTRAP:** Introduction to Grid System, Topography, Tables, Images, Themes, Forms, Inputs.[4]

6. **XML:** Introduction to XML, uses of XML, simple XML,XML keycomponents, DTD and Schemas, Well formed,using XML withapplication.XML, XSL and XSLT.Introduction to XSL, XML transformedsimple example, XSLelements, transforming with XSL . [3]

7. **PHP:** PHP: Starting to script on server side, Arrays, function and forms,advance PHPDatabases :Basic command with PHP examples, Connectionto server,creating database, selecting a database, listingdatabase, listing tablenames creating a table, inserting data,altering tables, queries, deletingdatabase, deleting data andtables, PHP myadmin and database bugs [7]

COURSE OUTCOMES(CO): The students are expected to:-

1. Understand the basic of internet.
2. Know the tags of HTML,JavaScript etc.
3. Understand the Bootstrap,XML,AJAX and PHP Programming.

4. To impart the design, development and implementation of Dynamic Web Pages.
5. To develop programs for Web using Scripting Languages.

Suggested Readings/Books:

1. HTML,CSS, JavaScript,Perl, Python and PHP, Wiley India Textbooks.
2. Internet and World Wide Web How to program, P.J. Deitel & H.M. DeitelPearson.
3. Steven Holzner,"HTML BlackBook",Dremtech press.
4. Deitel,Deitel, Nieto, and Sandhu: XML How to Program, Pearson Education
5. Web Applications : Concepts and Real World Design, Knuckles, Wiley-India
6. Online Free Resources from the Internet

BTCS-18953 Computational Biology

Internal Marks : 40 **L T P**
External Marks : 60 **3 0 0**
Total Marks : 100

Objective: The students will be able to develop software for predicting structure of protein, DNA and RNA and doing analysis of genetic and signaling pathways. They will also learn how to design drug computationally.

1. Biomolecular Structure Dynamics: Computational methods for pathways and systems biology, databases of metabolic pathways, Kyoto Encyclopedia of Genes and Genomes (KEGG), analysis of pathways, Glycolysis, signaling pathways, genetic pathways (4)

2. Gene Prediction: Computational gene mapping and gene hunting, genetic mapping, physical mapping, sequencing similarity search, gene prediction, mutational analysis, introduction to restriction mapping and map assembly, gene prediction methods, gene prediction tools, gene expression (5)

3. Gene Mapping: DNA double digest problem, multiple solutions to double digest problem, alternating cycles in colored graphs, physical maps and alternating Eulerian cycles, transformations in Eulerian cycles, partial digest problem, probed partial digest problem, homometric sets. Gene mapping, mapping with unique and non-unique probes, optical mapping, interval graphs, mapping with restriction fragment fingerprints, Lander-Waterman statistics, screening clone libraries, radiation hybrid mapping (7)

4. Computer Aided Drug Design: Computer aided drug design, methods of computer aided drug design, ligand design methods, docking algorithms and programs, drug design approaches, absorption, distribution, metabolism, and excretion (ADME) property prediction, computer based tools for drug design (5)

5. Soft Computation: Hidden Markov Model (HMM), Neural networks, machine learning, support vector machines, fuzzy logic, Evolutionary computing and genetic algorithms –application to data mining and bioinformatics, machine learning tools (MATLAB) (5)

6. Structure Databases: PDB and MMDB, structure file formats, visualizing information, advance structure modeling, Internal and external co-ordinate system and cylindrical polar co-ordinate system, potential energy calculations using semiempirical potential energy function, Electrostatic energy surface generation, three dimensional structure using dynamic programming methods, Molecular mechanics and dynamics (6)

7. RNA Secondary Structure and Perl Language: RNA secondary structure – combinatories, minimum free – energy structures, consensus folding, Unusual DNA structures, Perl language and Perl Programming (4)

Suggested Books

1. Computation Biology and Bioinformatics: Gene regulation by Ka- Chun Wong (2016). CRC Press, Taylor and Francis group, Science Publisher book, ISBN 9781498724975.

2. Bioinformatics Algorithm: An elective learning Approach, 2nd Edition, Vol. 1 by Phillip Compeau and Pavel Pevzner (2016). Active Learning Publishers, ISBN 10: 0990374610

3. Algebraic Stastics for Computational Biology edited by Lior Patcher and Bernd Sturmfels (2005), Cambridge Publishers ISBN: 10- 0521857007.

4. Computaional Molecular Biology: An Algorithm approach by P.A. Pevzner (2000). Publisher MIT Press, London, U.K, ISBN: 00- 032461

BTCS-18954 Computer Graphics

L	T	P
3	0	0

Internal Marks : 40
External Marks : 60
Total Marks : 100

Objectives of the course: • To provide comprehensive introduction about computer graphics system, design algorithms and two dimensional transformations.

• To make the students familiar with techniques of clipping, three dimensional graphics and three dimensional transformations, visible surface detection and shading.

Graphics Hardware: Basics of CRTs, Raster and Random Scan Displays, Graphics Input Devices, Graphics Output Devices, Applications of Computer Graphics.

Raster Scan Conversion Algorithms: Line Drawing Algorithms (DDA & Bresenham's), Circle Drawing Algorithms (Mid Point and Bresenham's).

Two-Dimensional Geometric Transformations: Basic Transformations, (Translation, Rotation and Scaling) Matrix Representation and Homogenous Coordinates, Shear and Reflection Transformations, Composite Transformations.

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

Windowing And Clipping: Window to viewport transformation, 2-D Clipping algorithms - Point Clipping, Line clipping (Cohen Sutherland), Polygon clipping (Sutherland Hodgeman, Weiler Atherton) and Text Clipping .

Elementary 3D Graphics : Basic Transformations: (Translation, Scaling, Rotation), Composite transformations.

Projections: Perspective Projection (Basic Principles, Mathematical Description and its Anomalies), Parallel projection (Basic Principles, Mathematical Description, Orthographics, Axonometric, isometric, dimetric, trimetric, Oblique projections, cavalier and cabinet projections).

Geometric Representation : Bezier and B-Spline Curves.

Visible Surface Detection Methods: Z-Buffer Method, Subdivision Algorithm, Floating horizon technique.

Shading : Gouraud and Phong Shading Algorithms.

Recommended Books:

1. Computer Graphics (Schaum Series) by Zhigang Xiang and Roy Plastock (MC Graw Hill)

2. Procedural Elements for Computer Graphics By David F Rogers, Second Edition, Tata McGraw-Hill Edition

3. Principles of Interactive Computer graphics: By W.M. Newman, R.Sproull, Tata McGraw-Hill Edition

4.. Computer Graphics Principles & Practice : By J.D. Foley, A. Van Dam, S. K. Feiner and J.F. Hughes, Second Edition, Pearson Education

5. Computer Graphics: By Donald Hearn, M. Pauline Baker

6. Computer Graphics Using OPEN GL: By F.S. Hill Jr.

Course Outcomes (CO) : At the end of the course student should have understanding of

1. Working of display devices
2. Implementation of line drawing algorithms, Circle drawing algorithms,
3. Implementation of 2D transformations of objects
4. Implementation of clipping algorithms
5. Implementation of 3D algorithms

BTCS-18955 Python Programming

Internal Marks : 40 **L T P**
External Marks : 60 **3 0 0**
Total Marks : 100

Course Objectives:
• Develop a basic understanding of the Python programming language.
• Learn various object types.
• Learn Numpy module for scientific computing.
• Learn to work with various types of data and convert it into meaningful information.
• Learn to visualize the data.

Introduction to Python: Python features, Type basics (Integer numbers, Complex numbers, Boolean numbers), Functions (Basic functions, local variables, global variables, variable scope, lambda functions),Loops, Flow Control Structures, Shared references, classes & objects.

Object types: Lists(Basic list operations, List iteration and comprehension, indexing, slicing, matrices), Dictionaries (Basic dictionary operations), Tuples (Basic tuple operations).

NumPybasics: Arithmetic with Numpy Arrays,Reshaping Arrays, Indexing, Slicing, Vectors and Matrices, Solving a Linear System.

Pandas: Creating Series objects,Series atrributes(index, values,dtype, isunique, ndim, shape, size),Series methods (sort_values, sort_index, count, describe, idxmax,idxmin,value_counts, head, tail), inplace parameter,DataFrame, Read data from csv file,Extracting columns from dataframe,Dataframe methods (sort_values, sort_index, astype, loc and iloc), Delete rows and columns from a dataframe, Broadcasting, Handling values (Null, Missing, Duplicate and Categorical), import excel file into pandas.

Visualization: UsingMatplotlib package, Creating Figures and Subplots, Creating charts (Line Chart, Scatter Chart, Bar chart, Pie chart, Box plot), Labels, Titles, Legends.

Recommended Books
1. Mark Lutz, “Learning Python”, 5th edition, O’Reilly.
2. Zed Shaw, “Learn Python the hard way”, 3rd edition, Pearson.
3. Eli Bresseert, “Scipy and Numpy”, O’Reilly.
4. William Mckinney, “Python for Data Analysis:Data wrangling with Pandas, NumPy, and Ipython”, 2nd edition, O’Reilly.
5. Phuong Vo.T.H, “Python: Data Analytics and Visualization”, Packt.

SIXTH SEMESTER

BTCS-18601 Compiler Design

Internal Marks: 40 **L T P**
External Marks: 60 **3 0 0**
Total Marks: 100

Objectives of the course

- To understand and list the different stages in the process of compilation.
- Identify different methods of lexical analysis
- Design top-down and bottom-up parsers
- Identify synthesized and inherited attributes
- Develop syntax directed translation schemes
- Develop algorithms to generate code for a target machine

Detailed contents

The aim is to learn how to design and implement a compiler and also to study the underlying theories. The main emphasis is for the imperative language.

Introduction: Phases of compilation and overview. Lexical Analysis (scanner): Regular languages, finite automata, regular expressions, from regular expressions to finite automata, scanner generator (lex, flex).

SyntaxAnalysis (Parser): Context-free languages and grammars, push-down automata, LL(1) gram-mars and top-down parsing, operator grammars, LR(O), SLR(1), LR(1), LALR(1) grammars and bottomup parsing, ambiguity and LR parsing, LALR(1) parser generator (yacc, bison)

Semantic Analysis: Attribute grammars, syntax directed definition, evaluation and flow of attribute in a syntax tree.

Symbol Table: Symbol table structure, symbol table attributes and management.

Run-time environment: Procedure activation, parameter passing, value return, memory allocation, and scope.

Intermediate Code Generation: Translation of different language features, different types of intermediate forms.

Code Improvement (optimization): Analysis: control-flow, data-flow dependence etc.; Code improvement local optimization, global optimization, loop optimization, peep-hole optimization loop optimization etc.

Code Generation: Register allocation and target code generation. DAG representation of programs and code generation from DAGs.

Course Outcomes

1. For a given grammar specification develop the lexical analyser.
2. For a given parser specification design top-down and bottom-up parsers.
3. Develop syntax directed translation schemes.
4. Develop algorithms to generate code for a target machine.

Books and References:

1. A. V. Aho, R. Sethi, and J. D. Ullman. Compilers: Principles, Techniques and Tools, Addison-Wesley.
2. C. Fischer and R. LeBlanc. Crafting a Compiler , Benjamin Cummings, 1991.
3. C. Fischer and R. LeBlanc. Crafting a Compiler in C.
4. Benjamin Cummings. A. C. Holub. Compiler Design in C , Prentice-Hall Inc., 1993.
5. Fraser and Hanson. A Retargetable C Compiler: Design and Implementation, Addison-Wesley.

BTCS-18602 Software Engineering

Internal Marks: 40 **L T P**
External Marks: 60 **3 0 0**
Total Marks: 100

Objective: To understand the basic and advanced concepts of software engineering.

Evolution and impact of Software engineering, software life cycle models: Waterfall, prototyping, Evolutionary, and Spiral models. Feasibility study, Functional and Non-functional requirements, Requirements gathering, Requirements analysis and specification

Basic issues in software design, modularity, cohesion, coupling and layering, function-oriented software design: DFD and Structure chart, object modeling using UML, Object-oriented software development, user interface design**.** Coding standards and Code review techniques.

Fundamentals of testing, White-box, and black-box testing, Test coverage analysis and test case design techniques, mutation testing, Static and dynamic analysis, Software reliability metrics, reliability growth modeling. [9]

Software project management, Project planning and control, cost estimation, project scheduling using PERT and GANTT charts, cost-time relations: Rayleigh-Norden results, quality management, ISO and SEI CMMI, PSP and Six Sigma. Computer aided software engineering, software maintenance, software reuse, Component-based software development. [12]

Suggested Books:

1. Roger Pressman, "Software Engineering: A Practitioners Approach,(6th Edition), McGraw Hill, 1997
2. Sommerville,"Software Engineering, 7th edition", Adison Wesley, 1996.
3. Watts Humphrey," Managing software process", Pearson education, 2003.
4. James F. Peters and Witold Pedrycz, " Software Engineering – An Engineering Approach", Wiley.
5. Mouratidis and Giorgini. "Integrating Security and Software Engineering– Advances and Future", IGP. ISBN – 1-59904-148-0.
6. Pankaj Jalote, "An integrated approach to Software Engineering", Springer/Narosa.

COURSE OUTCOMES (CO): The students are expected to:-

1. understand the software principles
2. understand the various phases of SDLC life cycle;
3. know the software project management;
4. understand the quality management like CMMI and Six Sigma; 5. Understand the implementation of software in 3GL and $GL environment.

BTCS-18604 Compiler Design Lab

Internal Marks: 30
External Marks: 20
Total Marks: 50

L T P
0 0 2

OBJECTIVES:-

The student should be made to:

- Be exposed to compiler writing tools.
- Learn to implement the different Phases of compiler
- Be familiar with control flow and data flow analysis
- Learn simple optimization techniques

PRACTICALS:-

1. Implementation of LEXICAL ANALYZER for IF STATEMENT
2. Implementation of LEXICAL ANALYZER for ARITHMETIC EXPRESSION
3. Construction of NFA from REGULAR EXPRESSION
4. Construction of DFA from NFA
5. Implementation of SHIFT REDUCE PARSING ALGORITHM
6. Implementation of OPERATOR PRECEDENCE PARSER
7. Implementation of RECURSIVE DESCENT PARSER
8. Implementation of CODE OPTIMIZATION TECHNIQUES
9. Implementation of CODE GENERATOR

OUTCOMES:-

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

- Implement the different Phases of compiler using tools
- Analyze the control flow and data flow of a typical program
- Optimize a given program

BTCS-18605 Software Engineering Lab

Internal Marks: 30	**L T P**
External Marks: 20	**0 0 2**
Total Marks: 50	

PRACTICALS:-

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 Software Requirement Specification Document, Design Documents and Testing Phase related documents for some problems

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 of a web site

Suggested Tools - Visual Paradigm, Rational Software Architect. Visio, ArgoUML, Rational Application Developer etc. platforms.

Elective – II

BTCS-18961 Mobile Application Development

Internal Marks: 40
External Marks: 60
Total Marks: 100

L T P
3 0 0

Unit I:

Introduction: Mobile operating system, Operating system structure, Constraints and Restrictions, Hardwareconfiguration with mobile operating system, Features: Multitasking Scheduling, Memory Allocation, FileSystem Interface, Keypad Interface, I/O Interface, Protection and Security, Multimedia features.

Unit II:

Introduction to Mobile development IDE's, Introduction to Worklight basics, Optimization, pages andfragments , Writing a basic program- in Worklight Studio, Client technologies, Client side debugging, Creating adapters, Invoking adapters from Worklight Client application, Common Controls, Using Java inadapters, Programming exercise with Skins, Understanding Apache Cordova, Offline access, Encrypted cachedeprecated, Using JSONStore

Unit III:

Understanding Apple iOS development, Android development, Shell Development, Creating Java MEapplication, Exploring the Worklight Server, Working with UI frameworks, Authentication, Push notification,SMS Notifications, Globalization, WebView overlay , Creating Authentication application: development forApple iOS by using a login module, Device Analytics, Worklight Server Administration

Unit IV:

Windows Phone: Introduction to Windows Phone, Architecture, memory management, communicationprotocols, application development methods, deployment.

Case Study: Design and development of Application using mobile application development platforms e.g. WorkLight, Kendo, Appcon, Xcode, Xpages

Unit V:

Android: Introduction to Android, Architecture, memory management, communication protocols, applicationdevelopment methods, deployment.

Case Study: Design and development of Application using mobile application development platforms e.g. WorkLight, Kendo, Appcon, Xcode, Xpages

Unit VI:

iOS: Introduction to iOS, Architecture, memory management, communication protocols, applicationdevelopment methods, deployment.

Case Study: Design and development of Application using mobile application development platforms e.g. WorkLight, Kendo, Appcon, Xcode, Xpages

Suggested Readings/Books:

1. Anubhav Pradhan, Anil V Deshpande, "Mobile Apps Development" Edition: I

2. Jeff McWherter, Scott Gowell "Professional Mobile Application Development", John Wiley & Sons, 2012.

3. Barry Burd, "Android Application Development All in one for Dummies", Edition:I

4. Teach Yourself Android Application Development In 24 Hours, Edition: I, Publication: SAMS

5. Neal Goldstein, Tony Bove, "iPhone Application Development All-In-One For Dummies", John Wiley &`Sons

6. Henry Lee, Eugene Chuvyrov, "Beginning Windows Phone App Development", Apress, 2012.

7. JochenSchiller,"Mobile Communications", Addison-Wesley, 2nd edition, 2004.

8. Stojmenovic and Cacute, "Handbook of Wireless Networks and Mobile Computing", Wiley, 2002, ISBN0471419028.

9. Worklight resources

BTCS-18962 Distributed Systems

L T P
3 0 0

Internal Marks: 40
External Marks: 60
Total Marks: 100

Course Objectives: The student should be made to:

- Understand foundations of Distributed Systems.
- Introduce the idea of peer to peer services and file system.
- Understand in detail the system level and support required for distributed system.
- Understand the issues involved in studying process and resource management.

UNIT-I

Characterization of Distributed Systems:Introduction, Examples of Distributed Systems, Resource Sharing and the Web, Challenges.System Models:Introduction, Architectural Models, Fundamental Models.[7]

UNIT-II

Time and Global States:Introduction, Clocks Events and Process States, Synchronizing Physical Clocks, Logical Time and Logical Clocks, Global States, Distributed Debugging.[8]

Coordination and Agreement:Introduction, Distributed Mutual Exclusion, Elections, Multicast Communication, Consensus and Related Problems.[7]

UNIT-III

InterProcess Communication:Introduction, The API for the Internet Protocols, External Data Representation and Marshalling, Client-Server Communication, Group Communication.[6]

Case Study: IPC in UNIX. Distributed Objects and Remote Invocation:Introduction, Communication between Distributed Objects, Remote Procedure Call, Events and Notifications, Case Study: JAVA RMI.[8]

Outcomes

1. Able to comprehend and design a new distributed system with the desired features.
2. Able to start literature survey leading to further research in any subarea.
3. Able to develop new distributed applications.

TEXT BOOK

Distributed Systems, Concepts and Design, George Coulouris, J Dollimore and Tim Kindberg, Pearson Education, 41″ Edition. 2009.

REFERENCE BOOKS

1. Distributed Systems, Principles and Paradigms, Andrew S. Tanenbaum, Maarten Van Steen, 2nd Edition, PHI.

2. Distributed Systems, An Algorithm Approach, Sukumar Ghosh, Chapman&Hall/CRC, Taylor & Fransis Group, 2007.

BTCS-18963 Machine Learning

Internal Marks: 40 **L T P**
External Marks: 60 **3 0 0**
Total Marks: 100

Course Objectives: The course is designed to understand the basic and advanced concepts of machine learning.

1. Introduction: Intelligent machines, Examples of applications in diverse fields, Data representation, Domain knowledge, Diversity of data, Forms of learning**.** (4)

2. Supervised learning: Learning from observations, Inductive learning, Ensemble learning, Evaluation of learning systems. (4)

3. Statistical learning: Machine learning and inferential statistical analysis, Descriptive statistics in learning techniques. (4)

4. Learning with Support Vector Machines: Linear discriminant functions for binary classification, Perception algorithm, Regression by support vector machines. (4)

5. Learning with Neural Networks: Neuron models, Activation function, Mathematical model, Network Architectures, Perceptrons. (3)

6. Fuzzy Interface systems: Cognitive uncertainty and fuzzy rule-base, Fuzzy quantification of knowledge, fuzzy rule-base and approximate reasoning, Neuro-fuzzy interface systems, Genetic fuzzy systems. (4)

7. Data clustering and Data transformations: Unsupervised learning, Clustring, Overview of basic clustering methods, useful data transformations. (3)

8. Decision tree learning: Classification decision tree, Measures of impurity for evaluating splits in decision trees Strength and weaknesses of decision tree approach (3)

9. Business Intelligence and Data Mining techniques and Applications: Introduction to analytics, Basic analytic techniques, Data warehousing and online analytical processing, intelligent information retrieval systems. (3)

10. Genetic algorithm for search optimization: Overview of genetics, Genetics on computers, Basic genetic algorithm. (4)

Course Outcomes:

1. Knowledge of different forms of learning;

2. Concepts of fuzzy systems, data clustering and decision trees;

3. Understanding of Business Intelligence and Data Mining techniques;

4. Knowledge of genetics and its algorithm.

Suggested Books:

1. M. Gopal, Applied Machine Learning, McGraw Hill Education (India) Private Limited, India.

2. Andreas Muller C. Sarah Guido, Introduction to machine learning using python, O' Reilly Media.

3. Michael Bowles, Machine Learning in Python: Essential Techniques for Predictive Analysis, Wiley and Sons.

4. Ethem Alpaydin**,** Introduction to Machine Learning, MIT Press.

BTCS-18964 Digital Signal Processing

L T P
3 1 0

Internal Marks: 40
External Marks: 60
Total Marks: 100

Module 1: Discrete-time signals and systems (6 hours)

Discrete time signals and systems: Sequences; representation of signals on orthogonal basis; Representation of discrete systems using difference equations, Sampling and reconstruction of signals - aliasing; Sampling theorem and Nyquist rate.

Module 2: Z-transform (6 hours)

z-Transform, Region of Convergence, Analysis of Linear Shift Invariant systems using z transform, Properties of z-transform for causal signals, Interpretation of stability in z-domain, Inverse z-transforms.

Module 2: Discrete Fourier Transform (10 hours)

Frequency Domain Analysis, Discrete Fourier Transform (DFT), Properties of DFT, Connvolution of signals, Fast Fourier Transform Algorithm, Parseval's Identity, Implementation of Discrete Time Systems.

Module 3:Designof Digital filters (12 hours)

Design of FIR Digital filters: ,Park-McClellan's method. Design of IIR Digital Filters: Butterworth, Chebyshev and Elliptic Approximations; Low-pass, Band-pass, Bandstop and High-pass filters. Effect of finite register length in FIR filter design. Parametric and non-parametric spectral estimation. Introduction to multi-rate signal processing.

Module 4: Applications of Digital Signal Processing (6 hours)

Correlation Functions and Power Spectra, Stationary Processes, Optimal filtering using ARMA Model, Linear Mean-Square Estimation, Wiener Filter.

Course Outcomes: At the end of this course, students will demonstrate the ability to

- Represent signals mathematically in continuous and discrete-time, and in the frequency domain.
- Analyse discrete-time systems using z-transform.
- Understand the Discrete-Fourier Transform (DFT) and the FFT algorithms.
- Design digital filters for various applications.
- Apply digital signal processing for the analysis of real-life signals.

Text/Reference Books:

1. S. K. Mitra, "Digital Signal Processing: A computer based approach", McGraw Hill, 2011.
2. A.V. Oppenheim and R. W. Schafer, "Discrete Time Signal Processing", Prentice Hall, 1989.
3. J. G. Proakis and D.G. Manolakis, "Digital Signal Processing: Principles, Algorithms And Applications", Prentice Hall, 1997.
4. L. R. Rabiner and B. Gold, "Theory and Application of Digital Signal Processing", Prentice Hall,1992.
5. J. R. Johnson, "Introduction to Digital Signal Processing", Prentice Hall, 1992.
6. D. J. DeFatta, J. G. Lucas andW. S. Hodgkiss, "Digital Signal Processing", John Wiley & Sons,1988.

BTCS-18965 Open Source Technologies

L T P

3 0 0

Internal Marks: 40
External Marks: 60
Total Marks: 100

Course Objectives: To provide the concepts of open source software and enable the students to learn Linux environment and implement the basics of MYSQL database

UNIT I

Open-Source Software Overview: Introduction–Need and Advantage of Open-Source Software ,Open Source Licensing Certification,OSS Development Model,Run a Free Software Project,Comparing OSS with other Software,OSS Licenses [8]

UNIT II

Open Source Operating System (LINUX): Installation of Linux (Red hat-CentOS): Hardisk Partitioning, Swap space, LVM, and Boot loader. Command Line: Basic File System Management Task, working with files, Piping and Redirection, working with VI editor, use of sed and understanding FHS of Linux.System Administration:Job management, Process Management, Mounting Devices and file system,Backup, Handling User Accounts, Groups and permission, Managing Software. Understanding Boot process and related files, Common kernel management Task [12]

UNIT III

Open Source Operating System (SHELL PROGRAMMING): Bash Shell Scripting, Executing Script, Working withVariables and Input, Using Control Structures, Handling signals, creating functions, working sed and gawk, working with web using shell script: Downloadingweb page,Converting Web page content to a text file, parsing data, working cURL.[8]

UNIT IV

Open Source Database And Application: MySQL: Configuring MySQL Server, working with MySQL Databases, MySQL Tables, SQL Commands – INSERT, SELECT, UPDATE, REPLACE, DELETE. Date and Time functions in MySQL. PHP,MySQL Application Development: Connecting to MySQL with PHP, Inserting data with PHP, Retrieving data with PHP.[8]

Course Outcomes:

1. Define the development model of OSS, and tell about the open-source licensing.
2. Demonstrate the installation of Linux by hard disk partioning and process of working with files
1. Analyze shell programming by working with variables, control structures and scripting
2. Develop Open Source Database by configuring MYSQL Server and connecting to MYSQL with PHP

Suggested Books:

5. D.S. Bedi, Strength of Materials, Khanna Book Publishing Company.
6. E.P. Popov, Mechanics of Materials-(SI Version), Prentice Hall India.
7. R.S Lehri and A.S. Lehri, Strength of Materials, Kataria and Sons.
8. S.S.Rattan, Strength of Materials, Tata McGraw Hill.

Elective - III

BTCS-18966 Parallel and Distributed Algorithms

Internal Marks: 40 **L T P**
External Marks: 60 **3 0 0**
Total Marks: 100

Course Objectives: The course objectives are: to learn parallel and distributed algorithms development techniques for shared memory and message passing models;to study the main classes of parallel algorithms; to study the complexity and correctness models for parallel algorithms.More specific: Learning the conceptual models for the specification and verification of parallel and distributed algorithms (P&D);Understanding the complexity models of P&D algorithms; Learning the development methods of efficient P&D algorithms using data partitioning and functional partitioning,parallelism exploitation for different process topologies, hiding communication latencies, load balancing; Development techniques for parallel algorithms;examples from the main classes: prefix algorithms, search, sort, selection, matrix processing, graph algorithms, lists, trees; Study of the main development techniques of classes of distributed algorithms: wave algorithms, mutual exclusion,topology establishment, termination, fault tolerance, leader election, genetic algorithms

1. Introduction of parallel and distributed algorithms:

Programming methods,Parallel and distributed systems,A language for algorithms presentation,Concurrency and synchronization,Atomicity,Barriers.

2. Complexity models of P&D algorithms:
Complexity of P&D algorithms Performance metrics,Complexity models: Work-depth, Foster, LogP,Algorithms development using shared variables,Languages for concurrent programming.Concurrent access to priority queues.

3. Development techniques for parallel algorithms:

Data parallelism,Prefix algorithms,Processing lists and matrices,Message passing,Algorithms development for PRAM models.Parallel search,Parallel selection,Algorithm development using message passing.

4. Development techniques of classes of distributed algorithms:

Distributed algorithms for mutual exclusion,Classification, complexity metrics,Lamport, Ricart-Agarwala, Roucairol-Carvalho, Maekawa, Submzuki Kasami, and Raymond algorithms.

5. Languages and libraries for distributed programming:

Logical clocks and ordering of events,Vector timestamps,Probe-echo messages,Termination of distributed programs,Termination detection in ring and graph topologies,Termination detection using message echoes: Dijkstra-Scholten,Detecting termination using mark messages,Huang algorithm,Algorithms for fault tolerant systems,Wave algorithms and leader election,Ring, tree, echo, and phase algorithms,Finn's algorithm,Leader election with wave algorithms,Token Based, non-token based algo Model, rationale, implementation,Transport problem.

Course Outcomes:

1. To reason about ways to parallelize a problem and be able to evaluate a parallel platform for a given problem

2. To develop an understanding of various basic concepts associated with parallel computing and distributed System environments.
3. To understand and explore the concepts with programming with MPI and MapReduce/
4. To become familiar with evaluation of online social networks and their potential
5. To gain experience in a number of different parallel computing paradigms including memory passing, memory sharing, data-parallel and other approaches.
6. To earn experience in designing and testing parallel computing solutions to programming problems.
7. tudents will identify the core concepts of distributed systems: the way in which several machines orchestrate to correctly solve problems in an efficient, reliable and scalable way.
8. Students will examine how existing systems have applied the concepts of distributed systems in designing large systems, and will additionally apply these concepts to develop sample systems.

Suggested Books:

6. Michel Raynal,Distributed Algorithms for Message-Passing Systems,Springer
7. S. K Basu, Parallel and Distributed Computing:Architectures and algorithms
8. Henri Casanova, Arnaud Legrand, Yves Robert, Parallel Algorithms,Chapman and Hall/CRC PublishedJuly17,2008Textbook-360Pages-119B/WIllustrations ISBN 9781584889458 - CAT# C9454
9. Nicola Santoro (Author),Design and Analysis of Distributed Algorithms (Wiley Series on Parallel and Distributed Computing Book 56) 1st Edition, Kindle Edition

BTCS-18967 Embedded Systems

Internal Marks: 40 **L T P**
External Marks: 60 **3 0 0**
Total Marks: 100

Introduction: Introduction to embedded systems and general purpose computer systems, architecture of embedded system, classifications, embedded microcontroller cores, embedded memories, applications and purpose of embedded systems, challenges and design issues in embedded systems, operational and non-operational quality attributes of embedded systems, elemental description of embedded processors. Examples of embedded systems. (12)

Technological aspects of embedded systems: Interfacing between analog and digital blocks, signal conditioning, digital signal processing, sub-system interfacing, interfacing with external systems, user interfacing , Design tradeoffs due to process compatibility, thermal considerations, etc. (l8)

Software aspects of embedded systems: Real time programming languages and operating systems for embedded systems. (6)

Text/Reference Books:

l. J.w. valvano, "Embedded Microcomputer System: Real time lnterfacing", Brooks/cole, 2000.
2. Jack Ganssle", The Art of Designing Embedded Systems, Newness, 1999.
3. V.K. Madisetti", VLSI Digital Signal processing "IEEE press (NY, USA), 1995.
4. David Simon," An Embedded Software Primer,' Addison Wesley,2000.
5. K'J. Ayala, "The 8051 Microcontroller: Architecture programming and Applications Penram In tl, 1996.

Course Outcomes:

At the end of the course students will demonstrate the ability to:
l. Suggest design approach using advanced controllers to real-life situations.
2. Design interfacing of the systems with other data handling/ processing system.

3. Appreciate engineering constraints like energy dissipation, data exchange speeds etc.

BTCS-18968 Microprocessor and Assembly Language Programming

Internal Marks: 40 **L T P**
External Marks: 60 **3 0 0**
Total Marks: 100

Objective/s: The course is intended to give students good understanding of internal architectural details and functioning of microprocessors.

Introduction:

Introduction to Microprocessors, history, classification, recent microprocessors.

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.

I/O memory interface:

Data transfer modes: Programmable, interrupt initiated and DMA. Serial & parallel data transfer, Detail study of 8251 I/O Processor & 8255 programmable peripheral interfaces.

Instruction set & Assembly Languages Programming:

Introduction, instruction formats & data formats, addressing modes, status flags, 8085 instructions, Data transfer operations, Arithmetic operations, Logical operations, Branch operations. Assembly language programming of 8085

Case structure & Microprocessor application:

Interfacing of matrix keyboards and seven segment LED display, Microprocessor controlled temperature system (MCTS), Study of traffic light system, stepper motor controller interface

Introduction to higher order microprocessors and microcontrollers:

Intel 8086 microprocessor, Intel 8051 microcontroller

COURSE OUTCOMES (CO):

The expected outcomes are:

1. Students should be able to solve basic binary math operations using the microprocessor.

2. Students should be able to demonstrate programming proficiency using the various addressing modes and data transfer instructions of the target microprocessor.
3. Students should be able to program using the capabilities of the stack, the program counter, and the status register and show how these are used to execute a machine code program.
4. Students should be able to apply knowledge of the microprocessor's internal registers and operations by use of a microprocessor based system design.
5. Students should be able to write assemble assembly language programs, assemble into machine a cross assembler utility and download and run their program on the training boards

Suggested books:

1. 8085 Microprocessor by Ramesh Gaonkar, PHI Publications.
2. Introduction to microprocessors by A.P. mathur, tata Mc. Graw Hill
3. Microprocessors and microcomputers by B.Ram, Dhanpat rai & sons

BTCS-18969 Ad-Hoc and Sensor Networks

L T P
3 0 0

Internal Marks: 40
External Marks: 60
Total Marks: 100

Course Objectives: The student should be made to Understand the design issues in ad hoc and sensor networks,Learn the different types of MAC protocols,Be familiar with different types of adhoc routing protocols and be expose to the TCP issues in adhoc networks.

UNIT I INTRODUCTION- Fundamentals of Wireless Communication Technology – The Electromagnetic Spectrum – Radio propagation Mechanisms – Characteristics of the Wireless Channel -mobile ad-hoc networks (MANETs) and wireless sensor networks (WSNs).concepts and architectures. Applications of Ad Hoc and Sensor networks. Design Challenges in Ad-hoc and Sensor Networks.[8]

UNIT II MAC PROTOCOLS FOR AD-HOC WIRELESS NETWORKS -Issues in designing a MAC Protocol- Classification of MAC Protocols- Contention based protocols,Contention based protocols with Reservation Mechanisms,Contention based protocols with Scheduling Mechanisms – Multi channel MAC-IEEE 802.11[9]

UNIT III ROUTING PROTOCOLS AND TRANSPORT LAYER IN AD HOC WIRELESS NETWORKS -Issues in designing a routing and Transport Layer protocol for Ad- hoc networks- proactive routing, reactive routing (on-demand), hybrid routing- Classification of Transport Layer solutions-TCP over Ad hoc wireless Networks.[9]

UNIT IV WIRELESS SENSOR NETWORKS (WSNS) AND MAC PROTOCOLS -
Single node architecture:hardware and software components of a sensor node–WSN Network architecture: typical network architectures-data relaying and aggregation strategies -MAC layer protocols: self-organizing, Hybrid TDMA/FDMA and CSMA based MAC- IEEE 802.15.4. [9]

Course Outcomes

Upon completion of the course, the student should be able to:

1.Explain the concepts, network architectures and applications of adhoc and wireless sensor networks.
2. Analyze the protocol design issues of adhoc and sensor networks.
3. Design routing protocols for ad-hoc and wireless sensor networks with respect to some protocol design

issues.

TEXT BOOK

Technical C. Siva Ram Murthy, and B. S. Manoj, "Ad Hoc Wireless Networks: Architectures and Protocols ", Prentice Hall Professional Reference, 2008.

REFERENCESBOOKS

1. Adhoc Wireless Networks — Architectures and Protocols, C.Siva Ram Murthy, B.S.Murthy, Pearson Education, 2004

2. Wireless Sensor Networks — Principles and Practice, Fei Hu, Xiaojun Cao,An Auerbach book, CRC Press, Taylor & Francis Group, 2010

3. Wireless Ad hoc Mobile Wireless Networks — Principles, Protocols and Applications, Subir Kumar Sarkar, et al., Auerbach Publications, Taylor & Francis Group, 2008.
4. Ad hoc Networking, Charles E.Perkins, Pearson Education, 2001.
5. Wireless Ad hoc and Sensor Networks — Protocols, Performance and Control, Jagannathan Sarangapani, CRC Press, Taylor & Francis Group, 2007, rp 2010.

SEVENTH SEMESTER

BTCS-18701 Artificial Intelligence

L T P
3 0 0

Internal Marks: 40
External Marks: 60
Total Marks: 100

Course Objectives:

1. Gain a historical perspective of AI and its foundations.

2. Become familiar with basic principles of AI toward problem solving, inference, perception, knowledge representation, and learning.

3. Investigate applications of AI techniques in intelligent agents, expert systems, artificial neural networks

1.Introduction- What is intelligence? Foundations of artificial intelligence (AI). History of AI; Problem Solving- Formulating problems, problem types, states and operators, statespace, search strategies. [6]

2.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, alpha-beta pruning [7]

3.Reasoning-Representation, Inference, Propositional Logic, predicate logic (first order logic), logical reasoning, forward chaining, backward chaining; AI languages and tools-Lisp, Prolog, CLIPS [5]

4.Planning- Basic representation of plans, partial order planning, planning in the blocks world, heirarchical planning, conditional planning [3]

5.Uncertainty - Basic probability, Bayes rule, Belief networks, Default reasoning, Fuzzy sets and fuzzy logic; Decision making- Utility theory, utility functions, Decisiontheoretic expert systems.[5]

6.Inductive learning - decision trees, rule based learning, current-best-hypothesis search, least-commitment search , neural networks, reinforcement learning, genetic algorithms. [5]

7.Communication - Communication among agents, natural language processing, formal grammar, parsing, grammar [5]

Course Outcomes:

1 .Understand the artificial intelligence
2. Know various search strategies
3. Understand AI Languages and tools
4. Know fuzzy logic concepts and their applications
5. Know the natural language processing
6. Understand the inductive learning concept**s.**

Suggested 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.George F. Luger, Artificial Intelligence: Structures and Strategies for Complex Problem Solving, Pearson Education Asia (2009) 6th ed.
5. ShivaniGoel, Express Learning- Artificial Intelligence, Pearson Education Asia (2013),1st ed.

BTCS-18703 Artificial Intelligence Lab

L T P
0 0 2

Internal Marks: 30
External Marks: 20
Total Marks: 50

Sno	Program Name
1	WAP to implement tower of Hanoi
2	WAP to implement BFS
3	WAP to implement DFS
4	WAP to implement cut and fail
5	Implementation of Family tree with predicates like father, mother, son,daughter, grandfather, aunt, uncle, cousin, ancestor for facts like male, female, parent, brother, sister.
6	WAP for medical diagnosis system of childhood diseases
7	WAP to implement 8 queens problem
8	WAP to implement tic tac toe using X and 0

Elective-IV

BTCS-18970 Network Security

Internal Marks : 40 **L T P**
External Marks : 60 **3 0 0**
Total Marks : 100

Course Objectives:

- To understand the fundamentals of Cryptography
- To acquire knowledge on standard algorithms used to provide confidentiality, integrity and authenticity.
- To understand the various key distribution and management schemes.
- To understand how to deploy encryption techniques to secure data in transit across data networks
- To design security applications in the field of Information technology

Basic Encryption And Decryption: Attackers and Types of threats, challenges for information security, Encryption Techniques, Classical Cryptographic Algorithms: Monoalphabetic Substitutions such as the Caesar Cipher, Cryptanalysis of Monoalphabetic ciphers, Polyalphabetic Ciphers such as Vigenere, Vernam Cipher, Stream and Block Ciphers.

Secret Key Systems: The Data encryption Standard (DES), Analyzing and Strengthening of DES, Introduction to Advance Encryption Standard (AES)

Public Key Encryption Systems: Concept and Characteristics of Public Key Encryption system, Introduction to Merkle-Hellman Knapsacks, Rivets – Shamir-Adlman (RSA) Encryption.

Hash Algorithms: Hash Algorithms, Message Digest Algorithms such as MD4 and MD5, Secure Hash Algorithms such as SH1 and SHA2.

Network Security: Network Security Issues such as Impersonation, Message Confidentiality, Message Integrity, Code Integrity, Denial of Service, Firewalls, DMZs, Virtual Private Networks, Network Monitoring and Diagnostic Devices.

Web Security: Web Servers, Secure Electronic Mail, Enhanced Email, Pretty Good Privacy, Public Key Cryptography Standards

Ethical Hacking: Introduction to Ethical Hacking, Terminology, Hackers, Crackers, and Other Related Terms, Hactivism, Threats, Hacking History ,Ethical Hacking Objectives and Motivations. .

Recommended Books:

1. Principles of Cryptography, William Stallings, Pearson Education
2. “Security in Computing (Second Edition)’, Charles P.Pfleeger, 1996, Prentice Hall International, Inc.
3. Cryptography & Network Security, Atul Kahate, TMH

BTCS-18971 Information Theory and Coding

L T P
3 0 0

Internal Marks: 40
External Marks: 60
Total Marks: 100

Information Theory: Introduction, Measure of information, Information content of message, Average Information content of symbols in Long Independent sequences, Average Information content of symbols in Long dependent sequences, Markov Statistical Model of Information Sources, Entropy and Information rate of Mark off Sources

Source Coding: Source coding theorem, Prefix Codes, Kraft McMillan In equality property – KMI Encoding of the Source Output, Shannon's Encoding Algorithm Shannon Fano Encoding Algorithm, Huffman codes, Extended Huffman coding, Arithmetic Coding, Lempel – Ziv Algorithm .

Information Channels: Communication Channels Channel Models, Channel Matrix, Joint probability Matrix, Binary Symmetric Channel, System Entropies, Mutual Information, Channel Capacity, Channel Capacity of :Binary Symmetric Channel, Binary Erasure Channel, Muroga,s Theorem, Contineuos Channels

Error Control Coding : Introduction, Examples of Error control coding, methods of Controlling Errors, Types of Errors, types of Codes, Linear Block Codes: matrix description of Linear Block Codes, Error Detection and Error Correction Capabilities of Linear Block Codes, Single Error Correcting hamming Codes, Table lookup Decoding using Standard Array.

Binary Cyclic Codes: Algebraic Structure of Cyclic Codes, Encoding using an (n-k) Bit Shift register, Syndrome Calculation, Error Detection and Correction.

Some Important Cyclic Codes: Golay Codes, BCH Codes. Convolution Codes: Convolution Encoder, Time domain approach, Transform domain approach, Code Tree, Trellis and State Diagram, The Viterbi Algorithm)

BTCS-18972 Distributed Operating System

L	T	P
3	0	0

Internal Marks : 40

External Marks : 60

Total Marks : 100

Objectives of the course: To examine the fundamental principles of distributed systems, and provide students hands-on experience in developing distributed protocols. While we still look at issues in distributed operating systems, this course will address distributed systems in a broader sense. Emphasis will be placed on communication, process, naming, synchronization, consistency and replication, and fault tolerance.

Introduction to distributed Systems: Definition and goals, Hardware and Software concepts, Design issues

Communication in Distributed System: Computer Network and Layered protocols, Message passing and related issues, synchronization, Client Server model & its implementation, remote procedure call and implementation issues, Case Studies: SUN RPC, DEC RPC

Synchronization in distributed systems: Clock synchronization and related algorithms, mutual exclusion, Deadlock in distributed systems

Processes and processors in distributed systems: Threads, system model, processor allocation, scheduling in distributed systems: Load balancing and sharing approach, fault tolerance, Real time distributed systems, Process migration and related issues

Distributed File Systems: Introduction, features & goal of distributed file system, file models, file accessing models, file sharing semantics, file caching scheme, file replication, fault tolerance, trends in distributed file system, case study.

Distributed Shared Memory: Introduction, general architecture of DSM systems, design and implementation issues of DSM, granularity, structure of shared memory space, consistency models, replacement strategy, thrashing

Naming: Overview, Features, Basic concepts, System oriented names, Object locating mechanisms, Issues in designing human oriented names, Name caches, Naming and security, DNS

Distributed Web-based Systems : Architecture, Processes, Communication, Naming, Synchronization, Consistency and Replication: Web Proxy Caching, Replication for Web Hosting Systems, Replication of Web Applications

Security : Introduction of Security in Distributed OS, Overview of security techniques, features, Need, Access Control, Security Management

Case Study : Java RMI, Sun Network File System, Google case study

Reference Books:

1. Distributed Operating Systems Concepts and Design, Pradeep K. Sinha, PHI

2. Distributed Systems: Concepts and Design by George Coulouris, Jean Dollimore,Tim Kindberg, Pearson

3. Distributed Operating Systems by Andrew S Tannebaum, Pearson

4. Distributed Computing by Sunita Mahajan & Seema Shah OXFORD

5. Distributed Systems: Principles and Paradigms by Andrew S Tanebaum, Maarten Van Steen, PHI

6. Distributed Computing, Fundamentals, Simulations and Advanced topics, 2nd Edition, Hagit Attiya and Jennifer Welch, Wiley India

Course Outcome:

After learning the course the students should be able to :

1. List the principles of distributed systems and describe the problems and challenges associated with these principles.

2. Understand Distributed Computing techniques, Synchronous and Processes.

3. Apply Shared Data access and Files concepts.

4. Design a distributed system that fulfills requirements with regards to key distributed systems properties.

5. Understand Distributed File Systems and Distributed Shared Memory.

6. Apply Distributed web-based system.

7. Understand the importance of security in distributed systems

BTCS-18973 Soft Computing

L T P
3 0 0

Internal Marks: 40
External Marks: 60
Total Marks: 100

Course Objectives: The course is designed to understand the basic areas of Soft Computing including Artificial Neural Networks, Fuzzy Logic and Genetic Algorithms. Provide the mathematical background for carrying out the optimization associated with neural network learning.

1.Introduction: Introduction to soft computing– Definition and importance, Evolution of soft computing, Usefulness and applications. [6]

2.Neural Networks: Model of an artificial neuron, Comparison of artificial neural network andBiological neural network. Neural network architectures. Learning methods–Hebbian,competitive, Boltzmann. Neural network models– Perceptron, Adaline and medaline networks,Single layer, Back propagation, Radial basis function network and multi-layer networks. [8]

3.Fuzzy Logic: Crisp and fuzzy sets.Fuzzy sets– Membership functions, Basic operations,Properties and fuzzy relations. Fuzzy rule based system–Linguistic hedges, Aggregation of fuzzyrules, Fuzzy inference system. Applications of fuzzy logic. [8]

4.Genetic Algorithms: Working principle– Crossover, Mutation, Encoding, Fitness function and Reproduction. Classification of genetic algorithm, Multi-objective genetic algorithm. Genetic Programming,Application of GA in search and optimization. [8]

5.Optimization Techniques: Simulated annealing, Particle Swarm Optimization, Bee Colony Optimization, Ant Colony Optimization, Tabusearch, Teaching-learning based optimization. [8]

Course Outcomes:

1.Comprehend the fuzzy logic and the concept of fuzziness involved in various systems andfuzzy set theory
2.Understand the concepts of fuzzy sets, knowledge representation using fuzzy rules, approximate reasoning, fuzzy inference systems, and fuzzy logic
3.To understand the fundamental theory and concepts of neural networks, Identify different neural network architectures, algorithms, applications and their limitations

Suggested Books:

1. S. Rajasekaran and G.A.V. Pai, "Neural Networks, Fuzzy logic and Genetic Algorithms",Prentice Hall of India.
2. Xin-She Yang , "Nature-Inspired Metaheuristic Algorithms", Luniver Press.
3. D.E. Goldberg, "Genetic Algorithms in Search and Optimization, and Machine Learning",Addison-Wesley.
4. V. Kecman, 'Learning and Soft Computing", MIT Press.
5. S.N. Sivanandam, S.N. Deepa, "Principles of Soft Computing", Wiley India.

BTCS-18974 Human Computer Interaction

L T P
3 0 0

Internal Marks: 40
External Marks: 60
Total Marks: 100

OBJECTIVES: The student should be made to:

- Learn the foundations of Human Computer Interaction
- Be familiar with the design technologies for individuals and persons with disabilities
- Be aware of mobile HCI
- Learn the guidelines for user interface.

UNITI:FOUNDATIONS OF HCI

The Human: I/O channels – Memory – Reasoning and problem solving; The computer: Devices – Memory – processing and networks; Interaction: Models – frameworks – Ergonomics – styles – elements – interactivity- Paradigms.

UNITII : DESIGN & SOFTWARE PROCESS

Interactive Design basics – process – scenarios – navigation – screen design – Iteration and prototyping. HCI in software process – software life cycle – usability engineering – Prototyping in practice – design rationale. Design rules – principles, standards, guidelines, rules. Evaluation Techniques – Universal Design.

UNIT III : MODELS AND THEORIES

Cognitive models –Socio-Organizational issues and stakeholder requirements –Communication and collaboration models-Hypertext, Multimedia and WWW.

Elective-V

BTCS-18975 Computational Number Theory

L T P
3 0 0

Internal Marks: 40
External Marks: 60
Total Marks: 100

Prerequisites

A student registering for this course is assumed to be equipped with rudimentary knowledge of discrete mathematical structures (groups, rings, fields), algorithms (design and analysis techniques), and probability. Note, however, that no prior acquaintance with number theory (elementary, analytic, or algebraic) is necessary for attending this course.

Syllabus

- **Algorithms for integer arithmetic:** Divisibility, gcd, modular arithmetic, modular exponentiation, Montgomery arithmetic, congruence, Chinese remainder theorem, primitive roots, quadratic residues, integer and modular square roots, prime number theorem, continued fractions and rational approximations.
- **Representation of finite fields:** Prime and extension fields, representation of extension fields, polynomial basis, primitive elements, normal basis, optimal normal basis, irreducible polynomials.
- **Algorithms for polynomials:** Root-finding and factorization, Lenstra-Lenstra-Lovasz algorithm, polynomials over finite fields.
- **Primality testing algorithms:** Fermat test, Miller-Rabin test, Solovay-Strassen test, Agrawal Kayal Saxena test.
- **Integer factoring algorithms:** Trial division, Pollard rho method, p-1 method, Continued fractions factoring method, quadratic sieve method, elliptic curve method.
- **Applications:** Algebraic coding theory, cryptography.

References

[1] V. Shoup, *A computational introduction to number theory and algebra*, Cambridge University Press.
[2] M. Mignotte, *Mathematics for computer algebra*, Springer-Verlag.
[3] I. Niven, H. S. Zuckerman and H. L. Montgomery, *An introduction to the theory of numbers*, John Wiley.
[4] J. von zur Gathen and J. Gerhard, *Modern computer algebra*, Cambridge University Press.
[5] R. Lidl and H. Niederreiter, *Introduction to finite fields and their applications*, Cambridge University Press.
[6] A. J. Menezes, editor, *Applications of finite fields*, Kluwer Academic Publishers.
[7] A. Das and C. E. Veni Madhavan, *Public-key cryptography: Theory and practice, Pearson Education Asia.*
[8] Abhijit Das, *A course in computational number theory*, CRC Press.

BTCS-18976 Speech and Natural Language Processing

Internal Marks : 40 **L T P**
External Marks : 60 **3 0 0**
Total Marks : 100

Course Contents

1. Basic Text Processing: Tokenization, Stemming
2. Language Modeling: N-grams, smoothing
3. Morphology, Parts of Speech Tagging
4. Syntax: PCFGs, Dependency Parsing
5. Distributional Semantics
6. Lexical Semantics, Word Sense Disambiguation
7. Information Extraction: Relation extraction, Event extraction
8. Text Summarization
9. Text Classification
10. Machine Translation

Reference Books

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

BTCS-18977 Parallel Architectures

Internal Marks: 40	**L T P**
External Marks: 60	**3 0 0**
Total Marks: 100	

Objectives of the course:-

- To develop an understanding of parallel computer architectures.
- To introduce various conditions of parallelism.
- Familiarization with different system interconnects networks.
- Familiarization with linear and non linear pipelining techniques.
- To know the concepts of multiprocessor systems and cache coherency.

Detailed contents:-

Parallel Computer Models – Elements of modern computers, Flynn's classification, clock rate and CPI, MIPS rate, throughput. Shared memory architecture: UMA, NUMA and COMA.

Program and Network Properties – Conditions of parallelism: Data and Resource Dependences, hardware and software parallelism. System interconnect architectures: static and dynamic connection networks.

Processors Hierarchy – Processors and coprocessors, RISC and CISC architectures**,** superscalar processors, VLIW architecture, vector processors.

Memory Hierarchy – Memory hierarchy**, i**nclusion, coherence and locality of reference, memory capacity planning, address space, mapping, TLB, paging, segmentation, replacement policies.

Pipelining Techniques – Linear pipeline processors: Asynchronous and synchronous models, clocking and timing control, speedup, efficiency and throughput. Nonlinear pipeline processors: Reservation and latency analysis, collision free scheduling. Instruction pipeline design and mechanism, branch handling techniques.

Multiprocessors– Multiprocessor system interconnects: characteristics, hierarchical bus, crossbar switch, multiport memory. Cache coherence problem and synchronization mechanism. Fault Tolerance, RAID.

Suggested books:-

1. "Advanced Computer Architecture", Parallelism, Scalability and Programmability, 3rd Edition, Kai Hwang and Naresh Jotwani, McGraw- Hill.
2. "Computer Architecture", Ian McLoughlin, Tata McGraw Hill Edition 2012.
3. "Computer System Architecture", 3rd Edition by M. Morris Mano, Pearson Edu. India.

Reference Books:-

1. "Computer Architecture and Organization", 3rd Edition by John P Hayes, McGraw Hill Edu.
2. "Computer Organization and Architecture: Designing for Performance", 10th Edition by William Stallings, Pearson Education.
3. "Computer Architecture: A Quantitative Approach", 5th edition by J.L. Hennessy and D. A. Patterson, Elsevier.

Course Outcomes:-

8. Understand the Concept of parallel architectures and its features.
9. Analyze the performance of different parallel architectures.
10. Understand the concept of instruction pipelining and design collision free scheduling.
11. Distinguish the performance of pipelining and non pipelining environment in a processor.
12. State and compare properties of multiprocessor systems and cache coherency protocols.

BTCS-18978 Data Mining

L T P
3 0 0

Internal Marks: 40
External Marks: 60
Total Marks: 100

Course Objectives: This course introduces basic concepts, tasks, methods, and techniques in data mining. The emphasis is on various data mining problems and their solutions. Students will develop an understanding of the data mining process and issues, learn various techniques for data mining, and apply the techniques in solving data mining problems using data mining tools and systems. Students will also be exposed to a sample of data mining applications.

Introduction to Data Mining:-Data Mining, Related technologies - Machine Learning, DBMS, OLAP, Statistics, Data Mining Goals,Stages of the Data Mining Process,Data Mining Techniques,Knowledge Representation Methods, Applications. [3]

Data Warehouse and OLAP: - What is data warehousing,The building Blocks: Defining Features, Data warehouses and datamarts,Overview of the components, Metadata in the data warehouse,Need for data warehousing, Data Warehouse and DBMS, Multidimensional data model, OLAP operations, Trends in data warehousing. [5]

Data preprocessing: - Data cleaning, Data transformation, Data reduction, Discretization and generating concept hierarchies. [2]

Data mining knowledge representation:- Task relevant data, Background knowledge, Interestingness measures**,**Representing input data and output knowledge, Visualization techniques. [2]

Attribute-oriented analysis: - Attribute generalization, Attribute relevance, Class comparison, Statistical measures. [3]

Concept Description and Association Rule Mining:- Association Rule Mining: Market basket analysis -basic concepts -Finding frequent item sets: Apriori algorithm -generating rules –Improved Apriori algorithm –Incremental ,ARM –Associative Classification –Rule Mining [4]

Data mining algorithms(classification and prediction): - What is classification and prediction? –Issues regarding Classification and prediction:Classification methods: Decision tree, Bayesian Classification, Rule based, CART, NeuralNetwork.Prediction methods: Linear and nonlinear regression, Logistic Regression.Introduction of tools such as DB Miner /WEKA/DTREG DM Tools [7]

Clustering:-Basic issues in clustering,First conceptual clustering system: Cluster/2,Partitioning methods: k-means, expectation maximization (EM),Hierarchical methods: distance-based agglomerative and divisible clustering,Conceptual clustering: Cobweb. [6]

Advanced techniques, Data Mining software and applications: Text mining: extracting attributes (keywords), structural approaches (parsing, soft parsing),Bayesian approach to classifying text,Web mining: classifying web pages, extracting knowledge from the web,Data Mining software and applications. [4]

Course Outcomes:

1. Demonstrate knowledge of key principles and techniques of data mining
2. Use statistical methods and visualization to explore and prepare data.

3. Describe the theoretical constructs and core processes of data mining
4. Demonstrate knowledge of various predictive modeling techniques.
5. Demonstrate knowledge of the emerging areas and applications of data mining
6. To gain experience of doing independent study and research.

Suggested Books:

1. J. Han, M. Kamber, "Data Mining Concepts and Techniques", Morgan Kaufmann.
2. M. Kantardzic, "Data mining: Concepts, models, methods and algorithms, John Wiley &Sons Inc.
3. Paulraj Ponnian, "Data Warehousing Fundamentals", John Willey.
4. M. Dunham, "Data Mining: Introductory and Advanced Topics", Pearson Education

BTCS-18979 Internet of Things

Internal Marks : 40 **L T P**
External Marks : 60 **3 0 0**
Total Marks : 100

Course Objectives:

- Able to understand the application areas of IoT
- Able to realize the revolution of Internet in Mobile Devices, Cloud & Sensor Networks
- Able to understand building blocks of Internet of Things and characteristics.

Introduction & Concepts: Introduction to Internet of Things, Physical Design of IoT, Logical Design of IoT, IoT Enabling Technologies, IoT Levels.

M2M to IoT: The Vision-Introduction, From M2M to IoT, M2M towards IoT-the global context, A use case example, Differing Characteristics. A Market Perspective- Introduction, Some Definitions, M2M Value Chains, IoT Value Chains, An emerging industrial structure for IoT, The international driven global value chain and global information monopolies.

M2M and IoT Technology Fundamentals: Devices and gateways, Local and wide area networking, Data management, Business processes in IoT, Everything as a Service(XaaS), M2M and IoT Analytics, Knowledge Management

IoT Architecture: State of the Art - Introduction, State of the art

IoT Reference Architecture: Introduction, Functional View, Information View, Deployment and Operational View, Other Relevant architectural views.

Domain Specific IoTs: Home Automation, Cities, Environment, Energy, Retail, Logistics, Agriculture, Industry, Health & Life Style.

Industrial Automation: Service-oriented architecture-based device integration, SOCRADES: realizing the enterprise integrated Web of Things, IMC-AESOP: from the Web of Things to the Cloud of Things

IoT Physical Devices & Endpoints: What is an IoT Device, Exemplary Device, Board, Linux on Raspberry Pi, Interfaces, and Programming & IoT Devices.

References:

1. Vijay Madisetti, Arshdeep Bahga," Internet of Things A Hands-On- Approach",2014
2. Adrian McEwen, "Designing the Internet of Things", Wiley Publishers, 2013
3. Daniel Kellmereit, "The Silent Intelligence: The Internet of Things". 2013

EIGHTH SEMESTER

BTCS-18801 Cloud Computing

L T P
3 0 0

Internal Marks: 40
External Marks: 60
Total Marks: 100

Course Objectives:

1. To introduce cloud computing, types of cloud services and enabling technologies.
2. To make them understand the role and usage of virtualization technologies.
3. To introduce cloud security issues and their resolution mechanisms.
4. To make them understand the features and usage of cloud platforms by studying the existing systems.

1. Overview of cloud computing : Cloud Computing Overview; Characteristics; Applications; Internet and Cloud; Benefits; Limitations; Challenges. Comparing grid with cloud and other computing systems, workload patterns for the cloud. Cloud computing leverages the Internet, Positioning cloud to a grid infrastructure, Elasticity and scalability. (7 hours)

2.Abstraction and Virtualization : Virtualization, Characteristics of Virtualization, Benefits of Virtualization, Virtualization in cloud computing,Types of Virtualization; Hardware Virtualization - full, partial, paravirtualization; Software Virtualization, Memory Virtualization; Storage Virtualization; Data Virtualization; Network Virtualization, Nested Virtualization; Hypervisor- Type-1, Type-2, Multitenancy, Types of tenancy, Billing and metering of services.(7 hours)

3.Cloud service delivery: Cloud service , Cloud service model architectures, Infrastructure as a service (IaaS) architecture, Infrastructure as a service (IaaS) details, Platform as a service (PaaS) architecture, Platform as a service (PaaS) details, Platform as a service (PaaS) , Examples of PaaS software, Software as a service (SaaS) architecture, Software as a service (SaaS) details, Examples of SaaS applications(5 hours)

4.Cloud deployment scenarios: Cloud deployment models, Public clouds, Hybrid clouds, Community, Virtual private clouds, Vertical and special purpose, Migration paths for cloud, Selection criteria for cloud deployment. (5 hours)

5.Cloud Storage – managed, unmanaged; Storage as a Service; Cloud Storage issues and challenges; Creating cloud storage system; Virtual storage containers; SAN, NAS, SAN vs. NAS (5 hours)

6.Security in cloud computing : Cloud security reference model, How security gets integrated , Cloud security , Understanding security risks, Principal security dangers to cloud computing, Virtualization and multitenancy, Internal security breaches, Data corruption or loss, User account and service hijacking, Steps to reduce cloud security breaches, Steps to reduce cloud security breaches, Identity management: Detection and Identity management, Benefits of identity, Encryption techniques, Encryption & Encrypting data , Symmetric key encryption, Asymmetric key encryption, Digital signature, What is SSL? (6 hours)

7.Cloud Computing Platforms:IBM Smart Cloud, Amazon Web Services, Google Cloud platform, Windows Azure platform, A comparison of Cloud Computing Platforms(5 hours)

Course Outcomes:

The students are expected to:-

1. Understand the concepts of Cloud Computing

2. Understand the various services being provided by clouds

3. Know the concepts of Cloud security

Suggested Books:

1.Raj Kumar Buyya, James Broberg, Andrezei M.Goscinski, Cloud Computing: Principles and paradigms, 2011

2.Michael Miller, Cloud Computing, 2008.

3.Essentials of Cloud Computing, K. Chandrasekaran

4.Cloud Computing, Pandey & Choudhary

5.Judith Hurwitz, Robin Bllor, Marcia Kaufman, Fern Halper, Cloud Computing for dummies, 2009.

6.Anthony T. Velte, Toby J. Velte and Robert Elsenpeter, Cloud Computing: A practical Approach, McGraw Hill, 2010.

7.Barrie Sosinsky, Cloud Computing Bible, Wiley, 2011.

8. Borko Furht, Armando Escalante (Editors), Handbook of Cloud Computing, Springer, 2010.

ELECTIVE-VI

BTCS-18981 Queuing Theory And Modelling

Internal Marks: 40 **L T P**
External Marks: 60 **3 0 0**
Total Marks: 100

Course Objectives: The aim of this course is to introduce various system modeling and simulation techniques, and highlight their applications in different areas. It includes modeling, design, simulation, planning, verification and validation. After learning the simulation techniques, the students are expected to be able to solve real world problems which cannot be solved strictly by mathematical approaches. This course begins by demonstrating the usefulness of simulation as a tool for problem solving in business, industry, government, and society.

UNIT I: RANDOM PROCESSES

Classification :Stationary process, Markov process, Poisson process, Discrete parameter, Markov chain, Chapman Kolmogorov equations, Limiting distributions.

UNIT II: QUEUEING MODELS

Markovian queues, Birth and death processes, Single and multiple server queueing models, Little‘s formula, Queues with finite waiting rooms , Queues with impatient customers : Balking and reneging.

UNIT III: ADVANCED QUEUEING MODELS

Finite source models : M/G/1 queue, Pollaczek Khinchin formula, M/D/1 and M/EK/1 as special cases, Series queues, Open Jackson networks.

UNIT IV: 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 V: TESTING HYPOTHESIS

Testing of Hypothesis: Formulation of Null hypothesis, critical 10 Hours region, level of significance, errors in testing, Tests of significance for Large and Small Samples, t-distribution, its properties and uses, F-distribution, its properties and uses, Chi-square distribution, its properties and uses, $\chi 2$ – test for goodness of fit, $\chi 2$ test for Independence

Course Outcomes:

1. Understand different methods for random number generation
2. Have a clear understanding of the need for the development process to initiate the real problem.
3. Have a clear understanding of principle and techniques of simulation methods informed by research direction.

4. Be able to discuss the simulation methods and select the suitable technique on the problems.

Suggested Books:

1. Gross, D., Shortle, J.F, Thompson, J.M and Harris. C.M., ―Fundamentals of Queueing Theory", Wiley Student 4th Edition, 2014.

2. Ibe, O.C., ―Fundamentals of Applied Probability and Random Processes", Elsevier, 1st Indian Reprint, 2007.

3. Sheldon M. Ross: Simulation 3rd Edition, Academic Press, 2002 •

4. M. Law and W. D. Kelton. Simulation Modeling and Analysis, 3rd Edition, McGrawHill, New York, USA, 1998

BTCS-18982 REAL TIME SYSTEM

L T P
3 0 0

Internal Marks: 40
External Marks: 60
Total Marks: 100

Course Objectives: It deals with issues in real time operating systems, importance of deadlines and concept of task scheduling. Student will be able to understand and design real time operating systems which are backbone of embedded industry.

Introduction: Definition, Issues in Real Time Computing, Structure of a Real TimeSystem. Task Classes

Characterizing Real Time Systems and Tasks: Introduction, Performance measuresfor real time systems: Traditional performance measures, Performability, Cost functionsand hard Deadlines

Task Assignment and Scheduling: Introduction, Classical Uniprocessor schedulingalgorithms: Rate Monotonic, EDF algorithm, Task assignment, Fault tolerant Scheduling .

Real Time Databases: Basic definitions, Real time Vs General Purpose databases, Main Memory databases, concurrency control issues, databases for hard real timesystems.

Real Time Communication: Introduction, Architectural Issues, Protocols: Contentionbased protocols, Token based protocols, Deadlines based protocols, Stop and GoMultihop protocol, The polled bus protocol, Hierarchical round robin protocol.

ExpectedCourse Outcomes:

1.Able to summarize the issues in real time computing

2.Able to explain and give examples of real time operating systems.

3. Able to solve scheduling problems and can apply them in real time applications in industry.

4.Design an RTOS and will be able to interpret the feasibility of a task set to accomplish or not.

5. Analyze the situation of fault occurrence and will be able to apply solutions accordingly.

Textbooks/References:

1. "Real Time Systems"-Liu Pearson Education

2. "Real Time Systems"-C. M. Krishna and Kang G. Shin

BTCS-18983 Data Analytics

L T P
3 0 0

Internal Marks: 40
External Marks: 60
Total Marks: 100

Course Objectives: This course will cover fundamental algorithms and techniques used in Data Analytics. The statistical foundations will be covered first, followed by various machine learning and data mining algorithms. Technological aspects like data management , scalable computation and visualization will also be covered. In summary, this course will provide exposure to theory as well as practical systems and software used in data analytics.

1. **Data Definitions and Analysis Techniques**: Elements, Variables, and DatacategorizationLevels of Measurement, Data management and indexing, Introduction to statistical learning and R-Programming.(7 hours)

2. Descriptive Statistics : Measures of central tendency, Measures of location of dispersions, Practice and analysis with R (8 hours)

3. **Basic Analysis Techniques:**Statistical hypothesis generation and testing, Chi-Square test, t-Test, Analysis of variance, Correlation analysis, Maximum likelihood test, Practice and analysis with R (10 hours)

4. Data analysis techniques: Regression analysis, Classification techniques, Clustering, Association rules analysis, Practice and analysis with R (10 hours)
5. Case studies and projects: Understanding business scenarios, Feature engineering and visualization (5 hours)

Course Outcomes:

After completing this course, you will learn how to

1. Find a meaningful pattern in data
2. Graphically interpret data
3. Implement the analytic algorithms
4. Handle large scale analytics projects from various domains
5. Develop intelligent decision support systems

Suggested Books:
The following text and reference books may be referred to for this course.

1. Probability & Statistics for Engineers & Scientists (9th Edn.), Ronald E. Walpole, Raymond H. Myers, Sharon L. Myers and Keying Ye, Prentice Hall Inc.

2. Data Mining and Analysis, Mohammed J. Zaki, Wagner Meira, Cambridge, 2012

3. The Elements of Statistical Learning, Data Mining, Inference, and Prediction (2nd Edn.), Trevor Hastie Robert

Tibshirani Jerome Friedman, Springer, 2014

4. An Introduction to Statistical Learning: with Applications in R, G James, D. Witten, T Hastie,and R. Tibshirani, Springer, 2013

BTCS-18984 Image Processing

L T P
3 1 0

Internal Marks: 40
External Marks: 60
Total Marks: 100

Digital Image Fundamentals-Elements of visual perception, image sensing and acquisition, image sampling and quantization, basic relationships between pixels – neighbourhood, adjacency, connectivity, distance measures.

Image Enhancements and Filtering-Gray level transformations, histogram equalization and specifications, pixel-domain smoothing filters – linear and order-statistics, pixel-domain

sharpening filters – first and second derivative, two-dimensional DFT and its inverse, frequency domain filters – low-pass and high-pass.

Color Image Processing-Color models–RGB, YUV, HSI; Color transformations– formulation, color complements, color slicing, tone and color corrections; Color image smoothing and sharpening; Color Segmentation.

Image Segmentation- Detection of discontinuities, edge linking and boundary detection, thresholding – global and adaptive, region-based segmentation.

Wavelets and Multi-resolution image processing- Uncertainty principles of Fourier Transform, Time-frequency localization, continuous wavelet transforms, wavelet bases and multi-resolution analysis, wavelets and Subband filter banks, wavelet packets.

Image Compression-Redundancy–inter-pixel and psycho-visual; Lossless compression –predictive, entropy; Lossy compression- predictive and transform coding; Discrete Cosine Transform; Still image compression standards – JPEG and JPEG-2000.

Fundamentals of Video Coding- Inter-frame redundancy, motion estimation techniques – full search, fast search strategies, forward and backward motion prediction, frame classification – I, P and B; Video sequence hierarchy – Group of pictures, frames, slices, macro-blocks and blocks; Elements of a video encoder and decoder; Video coding standards – MPEG and H.26X.

Course Outcomes:

At the end of the course, students will demonstrate the ability to:

- Mathematically represent the various types of images and analyse them.

- Process these images for the enhancement of certain properties or for optimized use of the resources.
- Develop algorithms for image compression and coding

Text/Reference Books:

1. R.C. Gonzalez and R.E. Woods, Digital Image Processing, Second Edition, Pearson Education 3rd edition 2008

2. Anil Kumar Jain, Fundamentals of Digital Image Processing, Prentice Hall of India.2nd edition 2004

3. Murat Tekalp , Digital Video Processing" Prentice Hall, 2nd edition 2015

BTCS-18985 Information Security & Cyber Law

Internal Marks : 40 **L T P**
External Marks : 60 **3 0 0**
Total Marks : 100

Course Objectives:

- To provide an understanding Computer forensics fundamentals
- To analyze various computer forensics technologies
- To provide computer forensics systems
- To identify methods for data recovery.
- To apply the methods for preservation of digital evidence.

Introduction to Internet, Cyber Space and threats, Computer Storage, Cell Phone / Mobile Forensics, Computer Ethics and Application Programs.

Electronic and Digital Signatures -Intellectual Property – Data Protection and Privacy. Need for cyber law and forensics, Jurisprudence of Indian Cyber Law.

Footprinting, WHOIS and DNS enumeration, network reconnaissance, Email spoofing, Email bombing, Data diddling, Denial of service attack, Virus / worm attacks- trojans and keyloggers, Internet time theft, Web jacking, Phishing-Smishing-Vishing-Identity theft, Cyber terrorism- use of encryption by terrorists, Human trafficking.

The Legal Perspective – The IT Act, Challenges faced by IT Act and its amendments, Sections Under IT Act-Section 43,65,66,67,68,69,70. Section relevant to cyber crime under IPC(Indian Penal Code).

Ehical hacking: Need, penetration testing: Information gathering tools like nmap, vulnearibility detection scanners like nessus, nexpose, information analysis and planning, attack and penetration tools like metasploit, results analysis and reporting.

Recommended Books

1. Cyber Security - Understanding cyber crimes,computer forensics and legal perspectives by Nina Godbole and Sunit Belapure.
2. System Forensics by Ankit Fadia.
3. hacking Exposed: network security secrets and solutions by Stuart mcclaure, Scambray and Kurtz. Tata Mc Graw hill.

Open electives odd semesters

BTCS-18991(Data Structures and Algorithms)

Internal Marks : 40 **L T P**
External Marks : 60 **3 0 0**
Total Marks : 100

Objectives of the course:

1 To impart the basic concepts of data structures and algorithms.
2 To understand concepts about searching and sorting techniques
3 To understand basic concepts about stacks, queues, lists trees and graphs.
4 To enable them to write algorithms for solving problems with the help of fundamental Data structures

Detailed contents:

Module1:

Introduction: Basic Terminologies: Elementary Data Organizations, Data Structure Operations: insertion, deletion, traversal etc.; Analysis of an Algorithm, Asymptotic Notations, Time-Space trade off. Searching: Linear Search & Binary Search Techniques and their complexity analysis.

Module2:

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

Module3:

Linked Lists: Singly linked lists: Representation in memory, Algorithms of several operations: Traversing, Searching, Insertion into, Deletion from linked list; Linked representation of Stack and Queue, Header nodes, Doubly linked list: operations on it and algorithmic analysis; Circular Linked Lists: all operations their algorithms and the complexity analysis. Trees: Basic Tree Terminologies, Different types of Trees: Binary Tree, Threaded Binary Tree, Binary Search Tree, AVL Tree; Tree operations on each of the trees and their algorithms with complexity analysis.

Module4:

Sorting and Hashing: Objective and properties of different sorting algorithms: Selection Sort, Bubble Sort, Insertion Sort, Quick Sort, Merge Sort, Heap Sort; Performance and Comparison among all the methods, Hashing. Graph: Basic Terminologies and Representations, Graph search and traversal algorithms . Minimum Spanning Trees : Prim‟s algorithm, Kruskul‟s algorithm

Suggested books:

1. "Fundamentals of Data Structures", Illustrated Edition by Ellis Horowitz, Sartaj Sahni, Computer Science Press. _

Suggested reference books:

1 Algorithms, Data Structures, and Problem Solving with C++", Illustrated Edition by Mark Allen Weiss, Addison-Wesley Publishing Company

2 "How to Solve it by Computer", 2nd Impression by R. G. Dromey, Pearson Education.

Course outcomes :-

1. For a given algorithm student will able to analyze the algorithms to determine the time and computation complexity and justify the correctness.

2. For a given Search problem (Linear Search and Binary Search) student will able to implement it.

3. For a given problem of Stacks, Queues and linked list student will able to implement it and analyze the same to determine the time and computation complexity.

4. Student will able to write an algorithm Selection Sort, Bubble Sort, Insertion Sort, Quick Sort, Merge Sort, Heap Sort and compare their performance in term of Space and Time complexity.

Student will able to implement Graph search and traversal algorithms and determine the time and computation complexity.

BTCS-18992 (Object Oriented Programming)

Internal Marks: 40 **L T P**
External Marks: 60 **3 0 0**
Total Marks: 100

Objectives of the course
The course will introduce standard tools and techniques for software development, using objectoriented approach.To understand Object Oriented Programming concepts and basic characteristics of C++.

Introduction
What is object oriented programming? Procedural Vs. Object-Oriented Programming , Basic Concepts and Principles of OOP

C++ Programming basics
Overview of C++, Program Structure, Exploring the Basic Components of C++ , Type Casting in C++, Operators in C++, Control Structures

Functions
Explore Functions , Describing Call by Value and Call by Reference , Inline Function, Overloading of Functions, String Library Functions, Recursive Functions, Friend Function.

Objects and Classes
Basics of Object and Class, Private and Public Members, Member Functions, Scope Resolution Operator, Constructors and their types, Destructors, Passing Objects as Function Parameters, Returning Objects from Functions.

Inheritance
Concept of inheritance, Derived class and base class, Types of Inheritance, Ambiguity and solution while implementing Multiple Inheritance.

Polymorphism
Concept of Polymorphism, Types of polymorphism, Function Overloading, Operator Overloading, Function Overriding.

Memory Management
Introduction to Pointers, Pointers and Objects, Dynamic Memory Management usingnew and delete operators, The this Pointer, pointer to object.

Templates and Exception Handling
Concept of Generic Programming, Function Template, Class Template, Exception handling mechanism, use of try, catch and throw keywords

Streams and Files
File Stream Operations, Opening and Closing a File, File Modes, File Pointers, Input Output Operations, Reading/Writing an object into file.

The concepts should be practiced using C++.

Suggested books

1.Lafore R., Object Oriented Programming in C++, Waite Group
2. E. Balagurusamy, Object Oriented Programming with C++, Tata McGraw Hill

Course Outcomes

After taking the course, students will be able to:

1.Recognise features of object-oriented design such as encapsulation, polymorphism,inheritance, and composition of systems based on object identity.

2.Apply these features to program design and implementation.

3.Design applications by using these object oriented concepts.

BTCS-18993 Computer Networks

Internal Marks: 40	**L T P**
External Marks: 60	**3 0 0**
Total Marks: 100	

Objectives of the course:-

- To develop an understanding of network architectures from the design and performance perspective.
- To introduce the students to the major concepts involved in local area networks (LANs), wide-area networks (WANs) and Wireless LANs (WLANs).
- To understand the requirement and use of various protocols at different levels of computer network architecture.
- To apply the knowledge of different network designs and various logical models of networking to solve problems of communication.
- To understand the basic concepts related to network security, firewall and cryptography.

Detailed contents:-

Data Communication Components:- Representation of data and data flow. Uses of computer networks. OSI and TCP/IP reference models. Various transmission media types. Network Topologies, Protocols and Standards, Wired and Wireless Networks. Techniques for Bandwidth utilization: Multiplexing- Frequency division, Time division.

Data Link Layer:- Fundamentals of Error Detection and Error Correction. Block coding, Hamming Code and CRC. Flow Control and Error control protocols - Stop and Wait, Sliding Window: Go back – N ARQ, Selective Repeat ARQ. Piggybacking.

Medium Access Sub Layer:- MAC Address. Multiple access protocols - Pure ALOHA, Slotted ALOHA, CSMA, CSMA/CD, and CDMA/CA. Binary exponential backoff algorithm. Collision free protocols.

Network Layer:- Logical addressing: IPV4, IPV6, CIDR, Subnet Mask, Default Gateway and DHCP. Circuit and Packet Switching. Routing algorithms: Flooding, Distance Vector, Link State. Congestion control policies, Leaky bucket and token bucket algorithms.

Transport Layer:- Port numbers and socket address. Process to Process Communication. Connection less and Connection oriented services. Transmission Control Protocol (TCP), User Datagram Protocol (UDP).

Application Layer:- Domain Name System (DNS), TELNET, EMAIL, File Transfer, Protocol (FTP), WWW, HTTP, SNMP, Firewalls, Basic concepts of Cryptography.

Suggested books:-

1. Data Communication and Networking, 4th Edition, Behrouz A. Forouzan, McGraw- Hill.
2. Computer Networks, 4th Edition, Andrew S. Tanenbaum, Pearson Education.

Reference Books:-

1. Internetworking with TCP/IP, Volume-I, Douglas E. Comer, Pearson Education.
2. Data and Computer Communication, 8th Edition, William Stallings, Pearson Education.
3. Computer Networking: A Top-Down Approach, James F. Kurose & Keith W. Ross, Pearson Education.

Course Outcomes:-

1. Explain the functions of the different layers of the OSI and TCP/IP Reference Models.
2. Understand the major concepts and devices involved in LAN and WAN.
3. Need and use of various error detection and correction methods.
4. Apply algorithms for medium access sub layer for maximum utilization of the bandwidth.
5. Configure networks, use of IP, MAC addresses, subnet mask, default gateway, ping and tracert.
6. Familiarization with application layer protocols: DNS, TELNET, FTP and HTTP etc.
7. Understand the importance of network security, firewall and cryptography.

Open electives for even semesters

BTCS-18994
Computer Organization & Architecture

Internal Marks: 40	**L T P**
External Marks: 60	**3 0 0**
Total Marks: 100	

Objectives of the course:

To expose the students to the following:

1 Understanding of its various functional units of computer system.
2 Microprocessors, instruction execution and assembly level programming.
3 Addition, Subtraction and Multiplication Algorithms.
4 Control Unit Microprogramming, Hardwired control unit
5 Memory hierarchy, mapping and memory system design
6 IO Modes: Program control IO, DMA, Interrupt initiated IO.
7 Basic concepts of pipelining, parallel processors and cache coherency.

Functional blocks of a computer: CPU, memory, input-output subsystems, control unit.
Microprocessor based system design, Introduction to 8085 architecture, addressing modes, instruction set and instruction execution cycle.

Data Representation and Binary Arithmetic: Signed number representation, addition, subtraction, Booth multiplication algorithm.

CPU control unit design: Hardwired and micro-programmed design approaches, Control Memory, RISC/CISC architecture.

Memory system design: Memory Hierarchy, memory organization, interleave memory, virtual memory, cache memory mapping techniques, and replacement algorithms, write policies.

IO Modes and Interfaces: Input-output subsystems, I/O device interface, I/O Modes – program controlled, interrupt driven and DMA.

Pipelining and parallel Processors: Basic concepts of pipelining, Introduction to parallel processors, Concurrent access to memory and cache coherency.

Suggested books:

1. "Computer System Architecture", 3rd Edition by M. Morris Mano, Pearson Education India.
2. "Fundamentals of Microprocessor and Microcontrollers", by B Ram, Dhanpat Rai Publications.

Reference Books:-

1. "Computer Architecture and Organization", 3rd Edition by John P Hayes, McGraw Hill Education.
2. "Computer Organization and Architecture: Designing for Performance", 10th Edition by William Stallings, Pearson Education.

Course outcomes:

1. Draw the functional block diagram of bus architecture of a computer and describe the function of the instruction execution cycle, interpretation of instructions, addressing modes.
2. Understand the concept of microprocessor based computer system architecture.
3. Familiarization with various addressing modes and assemble language programming.
4. Categorize memory organization and explain the importance of each element of memory hierarchy.
5. Identify and compare different methods for computer I/O mechanisms.
6. Understand the need of pipelining and parallel processing.

BTCS-18995 (Operating System)

Internal Marks: 40	**L T P**
External Marks: 60	**3 0 0**
Total Marks: 100	

Objectives of the course

To learn the fundamentals of Operating Systems.

1. To learn the mechanisms of OS to handle processes and threads and their communication
2. To learn the mechanisms involved in memory management in contemporary OS
3. To gain knowledge on distributed operating system concepts that includes architecture, Mutual exclusion algorithms, deadlock detection algorithms and agreement protocols
4. To know the components and management aspects of concurrency management

Detailed contents

Module 1:

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

Module 2:

Processes: Definition, Process Relationship, Different states of a Process, Process State transitions, Process Control Block (PCB), Context switching

Thread: Definition, Various states, Benefits of threads, Types of threads, Concept of multithreads

Process Scheduling: Foundation and Scheduling objectives, Types of Schedulers, Scheduling criteria: CPU utilization, Throughput, Turnaround Time, Waiting Time, Response Time;

Scheduling algorithms: Pre-emptive and Non pre-emptive, FCFS, SJF, RR; Multiprocessor scheduling: Real Time scheduling: RM and EDF.

Module 3:

Inter-process Communication: Critical Section, Race Conditions, Mutual Exclusion, Hardware Solution, Strict Alternation, Peterson's Solution, The Producer\ Consumer Problem, Semaphores, Event Counters, Monitors, Message Passing, Classical IPC Problems: Reader's & Writer Problem, Dinning Philosopher Problem etc.

Module 4:

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

Module 5:

Memory Management: Basic concept, Logical and Physical address map, Memory allocation: Contiguous Memory allocation – Fixed and variable partition– Internal and External fragmentation and Compaction; Paging: Principle of operation – Page allocation – Hardware support for paging, Protection and sharing, Disadvantages of paging.

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

Module 6:

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

File Management: Concept of File, Access methods, File types, File operation, Directory structure, File

System structure, Allocation methods (contiguous, linked, indexed), Free-space management (bit vector, linked list, grouping), directory implementation (linear list, hash table), efficiency and performance.

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

Suggested books:

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

Suggested reference books:

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

Course Outcomes

1. Create processes and threads.
2. Develop algorithms for process scheduling for a given specification of CPU utilization, Throughput, Turnaround Time, Waiting Time, Response Time.
3. For a given specification of memory organization develop the techniques for optimally allocating memory to processes by increasing memory utilization and for improving the access time.
4. Design and implement file management system.
5. For a given I/O devices and OS (specify) develop the I/O management functions in OS as part of a uniform device abstraction by performing operations for synchronization between CPU and I/O controllers.

BTCS-18996 (Database management Systems)

L T P
3 0 0

Internal Marks: 40
External Marks: 60
Total Marks: 100

Course Objectives:

6. To understand the different issues involved in the design and implementation of a database system.
7. To study the physical and logical database designs, database modeling, relational, hierarchical, and network models
8. To understand and use data manipulation language to query, update, and manage a database
9. To develop an understanding of essential DBMS concepts such as: database security, integrity, concurrency, distributed database, and intelligent database, Client/Server (Database Server), Data Warehousing.
10. To design and build a simple database system and demonstrate competence with the fundamental tasks involved with modeling, designing, and implementing a DBMS

Detailed contents:

Module 1

Database system architecture: Data Abstraction, Data Independence, Data Definition Language (DDL), Data Manipulation Language (DML).

Data models: Entity-relationship model, network model, relational and object oriented data models, integrity constraints, data manipulation operations.

Module 2:

Relational query languages: Relational algebra, Tuple and domain relational calculus, SQL3, DDL and DML constructs, Open source DBMS - MYSQL.

Relational database design: Domain and data dependency, Armstrong's axioms, Normal forms, Dependency preservation, Lossless design.

Query processing and optimization: Evaluation of relational algebra expressions, Query equivalence, Join strategies, Query optimization algorithms.

Module 3:

Storage strategies: Indices, B-trees, hashing.

Module 4:

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

Module 5:

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

Module 6:

Advanced topics: Object oriented and object relational databases, Logical databases, Web databases, Distributed databases, Data warehousing and data mining.

Course Outcomes

1. For a given query write relational algebra expressions for that query and optimize the developed expressions

2.For a given specification of the requirement design the databases using E_R method and normalization.

3. For a given specification construct the SQL queries for Open source DBMS –MYSQL

4. For a given query, optimize its execution using Query optimization algorithms

5. For a given transaction-processing system, determine the transaction atomicity, consistency, isolation, and durability.

6. Understand the isolation property, including locking, time stamping based on concurrency control and Serializability of scheduling.

Suggested books:

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

Suggested reference books

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